RLD'S ONLY MAGAZINE DEDICATED TO REAL ROBOTS

ROBOT
science & technology

The *Pathfinder* Fictio
The *Lost In Space* Fac
As Complex As You Wanna Be:
Your 1st Autonomous Robo

PREMIER COLLECTOR'S EDITION

HERE NOW: SILICON MAN

$4.95 US/$5.95 CAN

0 74470 93568 1 04

Bulk Rate
U.S. Postage
PAID
Permit No. 3
Sacramento,

The Nomad Scout

Join The Price/Performance Revolution

Nomad Scout

$3,999 without communications

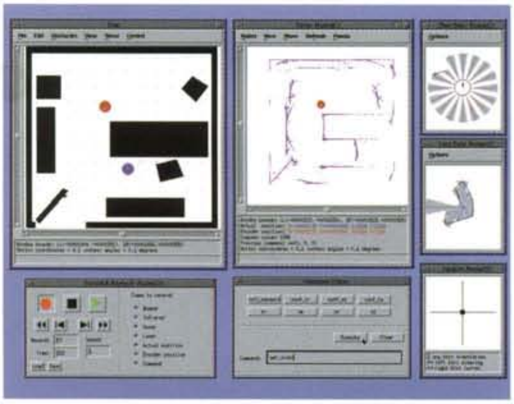
The Scout is an advanced low cost mobile robot system. The SymCentric™ drive system allows turning about geometric and sensing centroids. This, combined with 360° tactile and sonar sensing, greatly simplifies path planning and control algorithms.

Power

The Scout features a sophisticated power system: on board battery changing and monitoring, more than 24 hours of operating life, and AC interface for charging and direct operation.

Communications

High speed wireless ethernet with full TCP/IP support turns your Scout into a network node allowing seamless connectivity and control.



Software

The Scout is fully supported by the Nomadic Software Development Environment, the same full featured environment used by researchers with the industry standard Nomad 200 and 150 robots. This package features a fully functional multi-robot simulator, a GUI interface to simulated and real robots, advanced debugging and data recording tools, and interfaces to C, C++, and LISP.

Special RoboCup Offer:
Special Price/Special Options

Scout Pack

Make your own Scout RoboCup soccer team and take advantage of a special offer for five Scouts equipped with special RoboCup hardware options.

NOMADIC
TECHNOLOGIES, INC.

2133 Leghorn Street, Mountain View, CA 94043-1605, TEL 650.988.7200, FAX 650.988.7201, EMAIL nomad@robots.com, http://www.robots.com

PARALLAX INC

888.512.1024 (toll free)
916.624.8333
916.624.8003 fax
Monday-Friday 7 am to 5 pm PST

Tiny computers run PBASIC programs

BASIC Stamps are small computers programmed in Parallax BASIC (PBASIC), a simple programming language with powerful I/O instructions. The Parallax web site (http://www.parallaxinc.com) provides free software, manuals, and application notes.

BASIC Stamp® Modules

BS1-IC Module (#BS1-IC) $34
8 I/O lines; 80 PBASIC instr max; 2000 instr/sec; 2400 baud serial I/O; 14-pin SIP module. PBASIC language with I/O instructions including BUTTON, HIGH, INPUT, LOW, OUTPUT, POT, PULSIN, PULSOUT, PWM, REVERSE, SERIN, SEROUT, SOUND, and TOGGLE.

BS2-IC Module (#BS2-IC) $49
16 I/O lines; 500 PBASIC instr max; 4000 instr/sec; 50k baud serial I/O; 24-pin DIP module. Similar language as BS1-IC, plus DTMF, FREQOUT, SHIFTIN and SHIFTOUT, XOUT (X-10 powerline control), etc. I/O function have a higher resolution on the BS2-IC, due to its faster clock speed.

Using the PBASIC **HIGH** command and a 470 ohm resistor, BASIC Stamps can **electrify BLUE LEDs!** A stamper necessity! **(#27355) $8**

BASIC Stamp Bug (#27922) $129 (pictured above near Parallax Inc logo) The BASIC Stamp Bug is a walking robot with 6 legs that is controlled by the BASIC Stamp I interpreter chip. Antennas under the LED eyes attatch to switches which detect obstacles and inform the robot to maneuver around them.

Starter Kits

BASIC Stamp I Starter Kit (#27205) $99
BASIC Stamp II Starter Kit (#27203) $149
Starter Kits include BS1-IC or BS2-IC module, carrier board w/prototype area & 9V battery clip, manual, application notes, software,and free tech support.

NEW! BASIC Stamp Activity Board (#27905) $79 is used to learn and experiment with BS1-IC and BS2-IC modules. All components and current limit resistors are prewired to BASIC Stamp I/O pins. Board doubles as a "carrier board" with strip header access to I/O pins. Features include LEDs, pushbuttons, piezospeaker, an RC network for changing PWM into a smooth analog output, and an X-10 interface via RJ-11. Sample source code and power supply included!

Wireless RF Modules (#27924) $79
Designed by DVP and Parallax, these RF transmitter/receiver modules let you send RS-232 data (0-5 VDC) up to 1,000 feet away. Data transmission is most reliable with check sums and multiple string verification. Transmits at 303.825 MHz, a frequency reserved by the FCC for this type of use. The transmitter and receiver each have a 3-pin cable for connection to BASIC Stamp power, ground, and serial I/O pin. The optional power jack allows for a pluggable power solution if you don't want to use the included 3-pin cable for power.

2-line x 16 character Serial LCD Display (#27910) $54
Use the BASIC Stamp's SEROUT instruction (requires one I/O line, ground and power) to communicate with the Serial LCD display.

•**Australia** +61 67 722 777, +61 39 338 3306 •**Austria** +49 5232 8171 •**Belgium** +32 4 377 5151 •**Bulgaria** +3592 72 77 50 •**Brazil** +55 11 453 5588, +55 11 801 0045 •**Canada** (514) 336-9426 •**Czech Republic** +42 49 5813 252 •**Finland** +358 31 266 1885 •**France** +33 328 550 328 •**Germany** +49 5232 8171 •**Greece** +30 1 902 0115 •**Hungary** +36 1270 7680 •**India** +91 422 232 561 •**Ireland** +44 1 977 683 665, +35 31 821 5060 •**Israel** +972 3 498 543 •**Italy** +39 0542 55400 •**Japan** +81 3 3251 1779 •**Netherlands** +31 10 450 4949 •**New Zealand** +64 947 82323 •**Poland** +48 34 648 882 •**Slovak Republic** +42 7 580 2574 •**South Africa** +27 11 493 6242 •**Sweden** +46 431 41 00 88 •**Switzerland** +49 241 918 900 •**Taiwan** +886 2 647 1978 •**Thailand** ++66 2 739 1181 •**United Kingdom** +44 1 977 683 665 •**United States** 818 882-7621, 800 344-4539 **(Digi-Key)**, 800 831-4242 **(Jameco)**, 800 538-5000 **(JDR)**, 800 652-6733 **(Marlin P. Jones)**, 415 491-4600 **(Mondo-Tronics)**, 800 843-7422 **(Radio Shack Unlimited)**, 303 512-2060 **(TechAmerica)**

ROBOT science & technology

MICHAEL A. GREENE
publisher
editor-in-chief

DAVE ALBRIGHT
production manager
art director

TOM DURKIN
news & features editor

JAMES DI VITTORIO
PETER STRAIT
DOUG WINTER

JOHN PICCIRILLO, PH.D.
KRISTINE WILLS, M.D.
contributing editors

CONTRIBUTORS
Karl Lunt
Terry King
Ronni Katz
Robin Murphy, Ph.D.
The Iconoclast

ADVERTISING
Dave Albright 916.632.1000
ads@robotmag.com

EDITORIAL SUBMISSIONS
send manuscripts or outline to
Robot Science & Technology
2351 Sunset Blvd #170-253
Rocklin, CA 95765
editor@robotmag.com

Robot Science & Technology (ISSN 1096-4754) is published monthly at 2351 Sunset Blvd #170-253, Rocklin, CA 95765; 916.632.1000; Fax: 916.632.9216; info@robotmag.com.

Subscription inquiries: U.S. and Canada 888.510.7728; all other countries 916.632.1000. Issue price; $4.95 (U.S.) $5.95 (Canada). Subscription rates: 6 months $19.95 U.S., $29.95 Canada (U.S. funds). **Postmaster:** Send change of address to Robot Science & Technology 2351 Sunset Blvd #170-253, Rocklin, CA 95765.

ROBOT science & technology

Our Mission:
to immerse readers in 21st century robotics technology with in-depth reports on real robots and through hands-on adventures with home, classroom, and sport robots.

Contents

Cover Design:

The Real Silicon Man. Designed by Mike Greene & Dave Albright. The head of our silicon man incorporates images of a *BotBoard 2,* a microcontroller described in Karl Lunt's story, page 40. BotBoard supplied courtesy Marvin Green.

Trinity College FIRE FIGHTING Home Robot Contest

TWO $1,000 First Place Prizes

The 1998 Trinity College Firefighting Home Robot Contest will be held on the Trinity College campus Hartford, CT, on Sunday, April 19, 1998. This is the largest, public, true robotics competition held in the U that is open to entrants of any age, ability or experience from anywhere in the world. The goal of the contes to build a robot that can find and extinguish a fire in a house. The challenge for the entrants is to buil computerized (not radio-controlled) robotic device that can move through a model of a single floor of a hou detect fire (a lit candle), and then put out the flame.

Last year's weekend contest drew interest from people in all 50 states and 19 countries. Participants rang from college professors and engineers to elementary school students. There will be a Junior Division for th in High School and younger, and a Senior Division for everyone else. A cash prize of $1,000 will be awarded the top winner in each division, and additional cash prizes will go to other winners in those divisions.

Rules for the 1998 contest are $3.
Videotape of the 1997 contest is $25 (shipping included).
Send cash, check or money order – payable to TRINITY COLLEGE – to :

Jake Mendelssohn
190 Mohegan Drive
West Hartford, CT 06117

or download the rules from
www.trincoll.edu/~robot

If you have additional questions, contact: JMENDEL141@AOL.COM

Good luck with your robot. We'll see you in April!

Regional Contest Sites

Official regional contests have been established that operate under the same rules as the main contest in Hartford. Contact the sponsoring groups directly for more information about these regional contests.

Seattle Robotics Society, kevinro@nwlink.com, www.seattlerobotics.org
Southern Alberta Institute of Tech., craig.maynard@sait.ab.ca, www.robotgames.com
Ft. Worth IEEE Region 5, b.hayes@ieee.org, www.flash.net/~ieeeftw

Sponsored by MOTOROLA

Easy reading.
Cool robot.

F.E.A.R.

Fire Extinguishing Autonomous Robot

Robot Construction Instruction

Story and photos by ***Terry King***

Ever since I was a child, I have always been facinated by robots. I imagined a machine moving under its own power, acting under its own control. Movies and TV fueled my thoughts of creating my own robot.

While I was enjoying the imaginative entertainment, I constantly scrutinized the designs. What kind of batteries would supply enough power? What kind of propulsion would it take? How could the arms be designed to move, grab and lift objects? How would the sensors work? How could I make it *think*?

Now, many years since the *The Day the Earth Stood Still*, robots have become a part of our society, although not quite like *Robbie* – the robot from *Forbidden Planet*. Manufacturing plants and automobile factories have long used specialized robots. Additionally, robots are finding their way into fields such as medicine, underwater exploration, detonating explosive munitions and extraterrestrial exploration, like the Mars *Sojourner*.

In recent times, there has been a steady increase of interest in robotics by people in general. More and more, robot competitions have gained national and international attention. Good examples are contests such as the Trinity College Fire Fighting Robots Contest in Hartford, Conn. (www.trincoll.edu/~robot), Robot Wars (www.robotwars.com) in San Francisco and MIT-sponsored events.

In December of 1996, while surfing the Internet, I found out about the Trinity College contest. In short order, I was hooked. I researched everything I could about the contest – controllers, sensor systems and what other people were doing with robots in general.

Deciding that not only would I enjoy it, but that it would be a great father-son project, the creation of our fire fighting robot began. My son and I both agreed we wanted it to be autonomous. Thus, he named it *FEAR* – Fire-Extinguishing Autonomous Robot.

Perhaps you have considered constructing your own robotic marvel. Now, more than ever, is an excellent time to venture into this endeavor. Components and parts are available that make it possible for anyone to create a functional robot.

When building a robot, the first step involves planning. Before you pick up any tool or buy any materials, you should invest considerable thought into your robot's design.

What Do I Want It to Do?

Some builders have kept to a simple objective like, "Move around but avoid obstacles." Others have built robots to navigate mazes or search for a light source. Then there are those with more complex objectives.

Some have built robots to gather as many balls or other objects in the shortest time possible. Miniature robots have been engineered to play soccer as a team. In Robot Wars, your robot defends itself against other robots intent on its destruction - and attacks its opponents with equal zeal.

Our robot was designed entirely with the fire fighting contest in mind. *FEAR*'s purpose was to navigate the corridors without touching the walls, recognize the rooms, enter the rooms, and scan for presence of a flame. If your robot detected a flame, it was to approach the flame, extinguish it, exit the room, and return to the starting point. To put out the fire, *FEAR* pumped water through an adjustable spray nozzle.

> **TODAY, ANYONE CAN BUILD A FUNCTIONAL AUTONOMOUS ROBOT**

How Will It Behave?

Will it be autonomous or dead reckoning in its design?

First, consider autonomous robots. They collect data from sensory inputs and then evaluate what to do next. Their programming provides a set of behaviors in response to information made available to them. Their programs may have an overall objective to accomplish, but what it actually encounters during its mission will determine its exact actions.

Autonomous designs are meant to be adaptive in a variable environment. For example, an autonomous robot removed from one maze and placed in a new one will automatically adápt to the new maze.

Second, look at dead-reckoning designs. These robots use a preprogrammed set of instructions, which are carried out precisely. Each of its actions is predefined. It seeks its objective with intensity, oblivious to what is happening around it. While this design is effective in a known environment, it cannot deal with obstructions or changes. A dead-reckoning robot programmed for a particular maze, then removed to a new and different one, will instantly fail. However, if the robot's objective is constant, with all parameters remaining the same every time, the dead-reckoning approach can offer precision behavior.

Whichever design you choose, you must make the decision early on. Autonomous designs rely on good sensory information and lots of it. Dead-reckoners need good encoder information to determine precise distances traveled.

FEAR *with its hood up: Note the adjustable spray nozzle on the robot's water cannon and the water tank mounted on the turret. See the onboard controller under the turret.*

What Will Control It?

You must select the controller, or microprocessor, that will be the "brains" of your robot. There are a variety of devices that lend themselves to robotics. Some controllers are packaged with the necessary peripherals, such as motor drivers, digital and analog I/O (input/output) ports, and onboard memory. Their programming formats include BASIC, variations of BASIC, assembly, C, or variations of C.

Popular examples of these controllers are the BASIC Stamp, invented at Parallax, and the Handy Board, invented at the MIT Media Lab. Both are loaded with excellent features ideal for mobile robots.

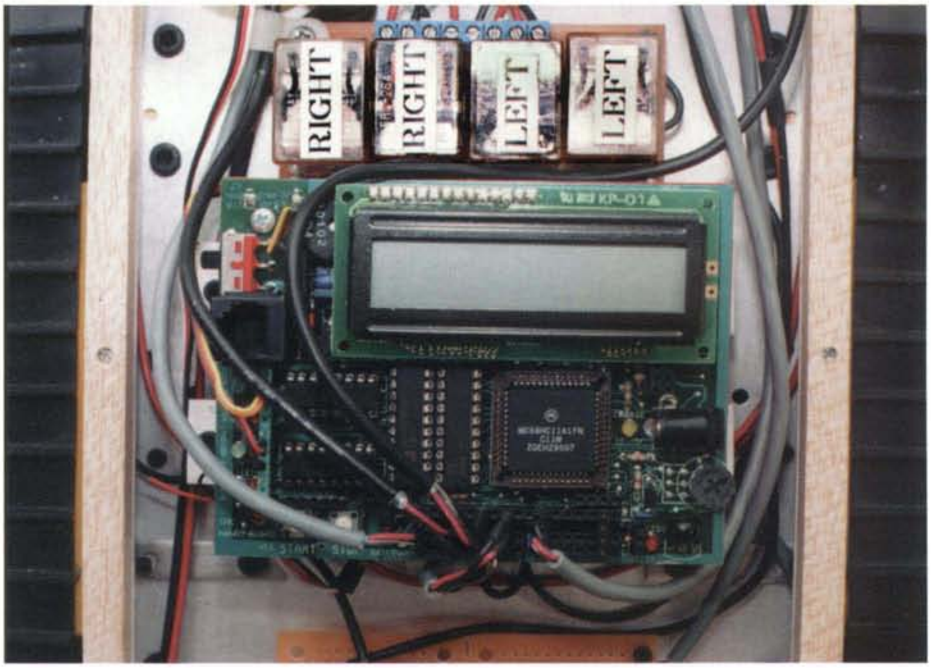
To build *FEAR*, we decided to use the Handy Board. It came with a stout battery pack, LCD (liquid crystal display) screen, 32K of memory, digital and analog I/O ports, four 1-amp motor ports, and a software package called Interactive C.

My employer, American Communications Network of Bakersfield, CA (www.acncommunications.com), generously sponsored the purchase.

Tracks? Wheels? Legs?

Once you have your controller, you must next decide on the type of drive system best suited for the terrain your robot will traverse. Will the topography be rough and uneven? Smooth? Slippery?

Track, or Caterpillar-type, treads can handle just about any kind of terrain. They also greatly simplify the problem of steering. Having one tread go forward with the other reversed makes snappy turns. But that kind of maneuverability comes at a price. Track designs take considerably more power than wheels because of the increased points of friction. As noted above, good sensory information is important here to ensure that a 90-degree turn is *actually* 90 degrees.



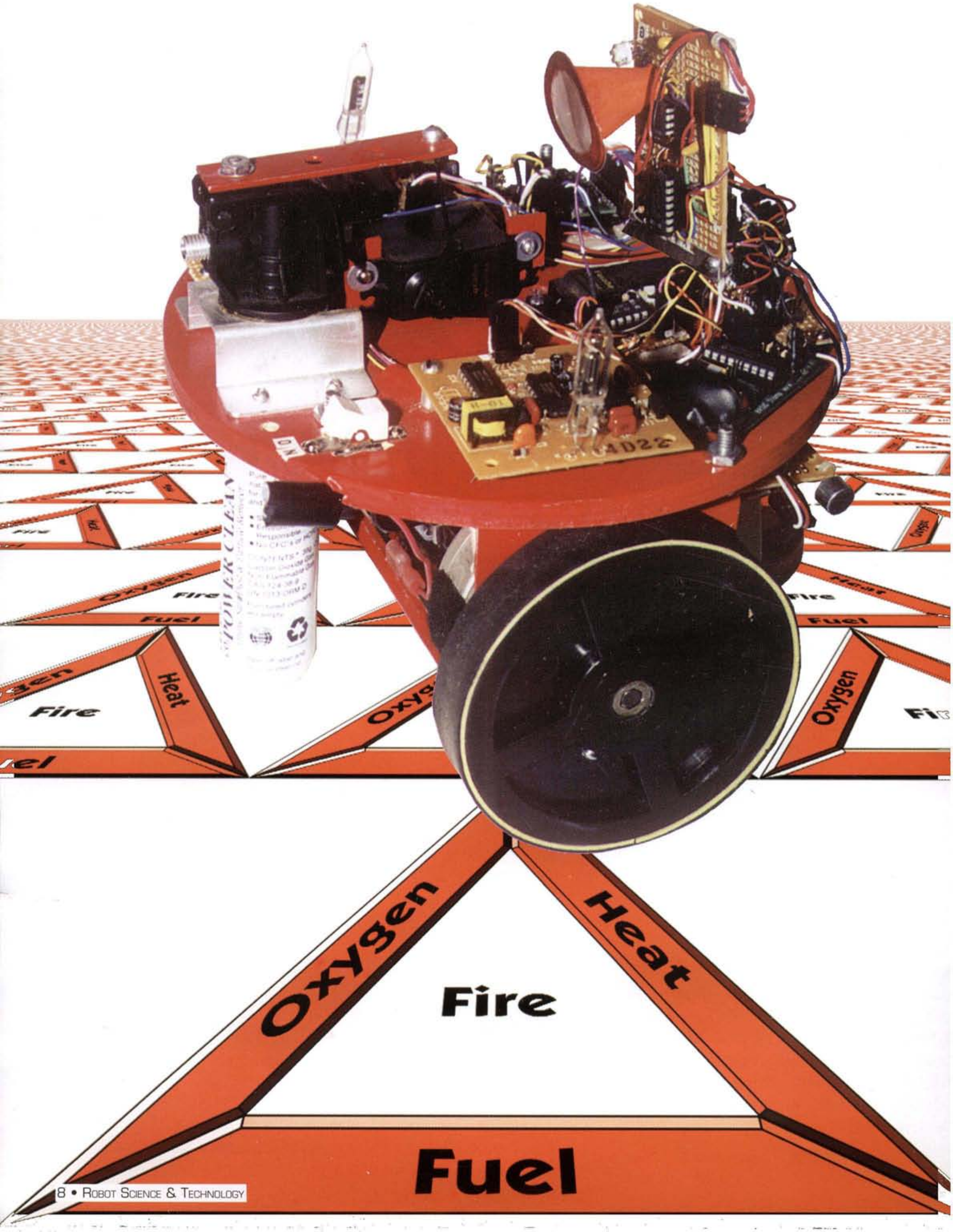
The programmable Handy Board, the brains of FEAR, *gave our firefighting robot the autonomous ability to find and extinguish a small fire. Note the LCD screen mounted on the board and the four 2-amp motor ports just above it.*

One problem with industrially available RC performance vehicles is the large amount of current they consume. Some people have had success with the chassis of radio-controlled cars made by Traxxas (www.traxxas.com). Toy stores or department stores like Wal-Mart often sell scaled-down, medium-quality, treaded earth-moving toys. Look for one that already has separate right and left drive motors.

Wheels are more energy-efficient but may have traction problems under some conditions. Four-wheel designs make for a stable robot, but on the downside, steering is complicated. (See *Navigating Mobile Robots: Systems and Techniques* by J. Borenstein, H. R. Everett and Liqiang Feng.) Two-wheel designs with two drive motors offer steering response like that of tracks, with the added advantage of requiring less energy to turn. On the other hand, they require some sort of caster or skids to balance them. Under smooth, ideal conditions, wheels can prove to be very effective.

Walking robots bring about a very different set of design hurdles. Multi-legged configurations must

continued on page 22


Oxygen
Heat
Fire
Fuel

Fire Science Meets Robot Technology

True Confessions of a Hack Designer

by John Piccirillo

Here are the nuts and bolts details of how I designed and built a robot for the Fire Fighting Home Robot Contest.

And I won't hide the process, including my mistakes.

I wish I could say that I followed a top-down approach from beginning to end. But unfortunately, I'm not that good an engineer. Actually, I relied partly on the robot's behavior and on extensive testing to guide the development process, (especially to find the failure modes).

Very briefly, the basic requirement is to build a robot that will navigate a known floor plan in an 8′ x 8′ square foot robot house, find a lit candle, and extinguish it in a minimum amount of time.

The five basic systems required are: the navigation platform, the flame detector, the extinguisher, the computer, and the control software.

A bonus is given for starting the robot with a buzzer tone so I'll include a sound activation unit. In summary then, here are my top level design requirements.

1. A platform to navigate a known floor plan. Max size was a 12″ cube. To allow for navigating errors, smaller is better.

2. A sensor to reliably detect a standard candle flame at a max distance of 4′. The base of the candle flame can vary from 6″ to 8″ above the floor.

3. A candle extinguisher.

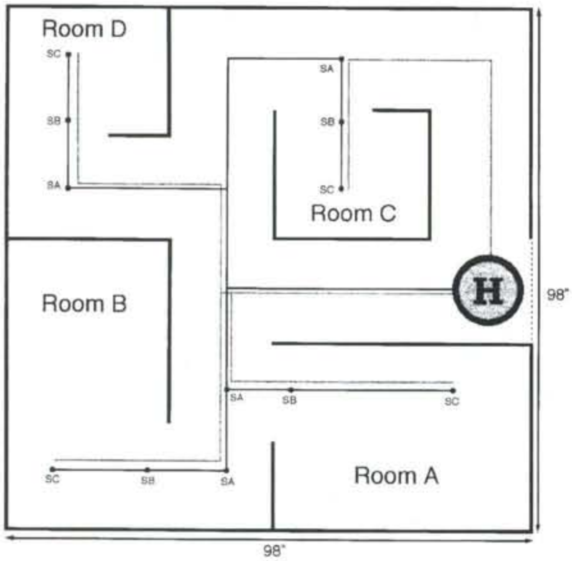
4. A single-board computer that has a simple development environment.

5. Software: an optimal search strategy in a higher-level language. A bonus is given for returning to the starting point after the candle is put out, so the software needs return-path tables.

6. A sound activation circuit to work with a buzzer.

As I worked, a compact and reliable robot emerged, but only after several missteps. It helped that I had

1996 Floor Plan



These dimensions are approximations and are NOT 100% accurate and that's why the numbers don't add up exactly. Welcome to the real world!

experienced a previous fire fighting contest, and had watched video tapes of past events.

As a robot, my new robot, Marv, is different in every system from my first fire fighting robot, which competed in 1994's contest.

Background

My original plan was to build on what I had done with Marv-94. It was 10" in diameter (with two layers), used a motor and gearbox from an old Big Trac toy, a CdS visible light sensor, a car windshield washer pump to squirt water on the candle, and wheel encoders to measure the distance traveled. It searched for the candle in a fixed order among the four available rooms.

Coming to a doorway, Marv-94 turned, entered and scanned the light sensor with a stepper motor. If no candle was found, Marv would advance further in the larger rooms and search again before re-tracing its steps and trying the next room. When the candle was found, the pump sprayed water on the flame.

This technique worked fairly well and Marv took fourth place. However, making Marv really reliable required a few tricks.

1. The scanning light sensor looked for an increase, followed two steps later by a decrease in light level. It also compared the light level measured two steps earlier with the current measurement, rather than with some preset light level. These precautions eliminated the common problem of unknown and non-uniform backgrounds.

2. A spray tip from a garden insecticide sprayer was used to set Marv's water stream. During discharge the stream was swept from side-to-side a little to compensate for pointing errors. I made the sprayer more effective with an additive purchased from a fire protection services business (look under Fire Extinguishers in the yellow pages).

3. Marv's navigation was atrocious and he frequently collided with walls and got stuck. After a long talk with him, I added a rear bumper with two hemispherical bumps (a roll-on deodorant ball cut in half) at either end. Every now and then Marv would back up, crash into a wall and thereby square itself - comical to watch, but effective.

4. Marv used "kill assessment." After Marv sprayed a candle it would re-scan the light sensor to confirm that the flame was out. If it was not, Marv moved closer and sprayed again.

With these strategies, Marv put the candle out in two out of three runs.

EVERY NOW AND THEN MARV WOULD BACK UP, CRASH INTO A WALL AND THEREBY SQUARE ITSELF - COMICAL TO WATCH, BUT EFFECTIVE.

The New Marv

For the new Marv I wanted a smaller platform, and aimed for an 8" diameter. I also wanted to add UV flame detectors. These highly sensitive devices use a part of the UV spectrum (185 to 260 nanometers) called the solar blind. Since this is not present in most light sources the sensor can detect a candle flame by reflected light without being affected by background lighting. This allows it to see a candle *even if it's not directly visible*. This was a crucial feature of my design, as I'll describe below.

From my prior experience, I knew that most navigation problems result from the inability to make *exact* 90 degree turns. An error as small as a couple of degrees soon has the robot crashing into walls. Therefore I wanted to make the minimum number of turns.

One way to do this is to check a room for a lit candle without turning to enter it. So Marv traveled down a hall, stopped in front of a doorway to determine if a candle was present (using the UV sensor, which has an at-large field-of-view). No turns

continued on page 2

Why A Magazine About Robots *Now*?

Our unique magazine reaches a highly focused readership. Two years of market testing and trade research has proved that tens of thousands of people are already building robots: home robots, classroom robots and sport robots. Our audience, hungry for information, has been clamoring for a dedicated magazine. The ingredients we promise are: commitment to excellence, respect for the intelligence of our readers, and enthusiasm for robotics.

We are multiplying the size of the robotics market

by reaching an audience who has not yet seen a magazine about real robots.

What is Robot Science & Technology?

RS&T is the only dedicated gateway to this new rapid-growth market.

You're enjoying a fresh new kind of magazine that tens of thousands of consumers have never seen before: a unique monthly resource to help them comprehend the robotic applications that are playing an increasingly meaningful role in our lives. We write to a level that any technically-oriented consumer can read to get a hands-on understanding of one the most powerful technologies of the 21st century.

Our expert writers help readers build affordable, useful robots for home, classroom and competition use. Every month we focus on:

Highlights of Recent Robot Competitions and Events

(There are over a hundred major competitions each year, many drawing thousands of spectators and dozens of robotic competitors.)

Preparation Tips for Upcoming Contests

(How to build, how to compete, how to win.)

Resources for Educators, Small Robotics-Related Businesses and Robot Clubs

(College engineering and computer science departments, even high schools, are *filling* classes in artificial intelligence, robotic behavior and robot design.)

Plus: What's New in Humanoids, Space, Exploration, Police & Military Robots

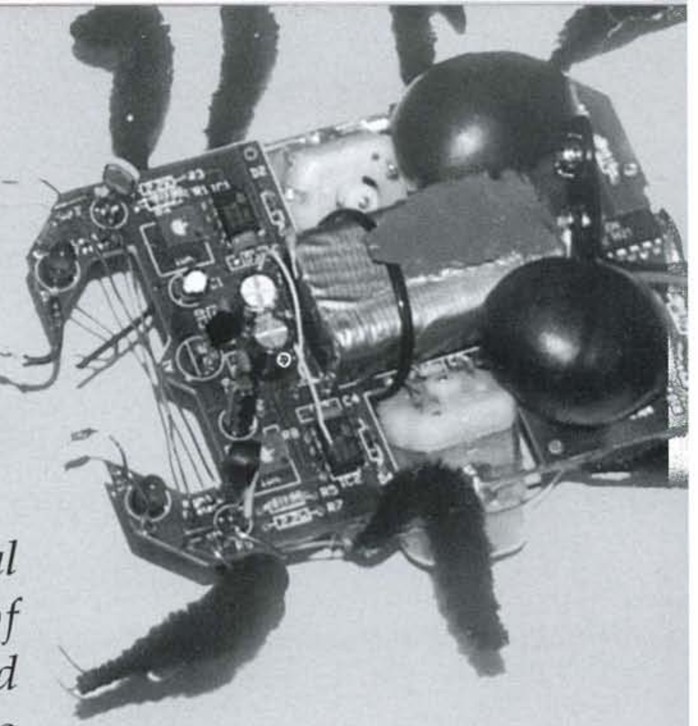
Do you remember how a few techno-nerds changed our world in the '70s & '80s by using computers?

THE PERSONAL ROBOTICS EXPLOSION IS HAPPENING NOW, in schools, clubs and garages across the land.

Who Reads RS&T?

Educators: Our hands-on projects make learning interesting and concrete.
Students: The brightest, most curious students get more from extracurricular reading.
Hobbyists: Building robots is a hands-on hobby, perfect for tech-heads.
Researchers: Our articles offer a fresh perspective.
Techies: In the New Millennium, robotics will be one of the most promising job opportunities.
Consumers: They're looking for something new, something exciting, something very *Wow*!

CYBUG: A *Real* Computer Bug



CYBUG 1 is an exciting new educational robot kit. It combines elements of electronics, robotics, mechanics and ecology in a unique and interesting package, but it uses no microprocessor and requires no programming. It's instructional, easy to build, and fun to modify and customize.

The CYBUG acts like a living organism, with behaviors and instincts designed into its circuitry. This "robo-organism" is nocturnal (most active at night) and can be configured to be phototropic (light-seeking) or photophobic (light-avoiding). It has a pair of "feelers" that allow it to avoid obstacles or edges.

A quick adjustment of its potentiometer transforms a CYBUG from a fast-moving, aggressive "predator" to a slow but energy-efficient "herbivore."

No two CYBUGs are exactly the same, and you can come to recognize the unique characteristics of each individual.

One of the most fascinating attributes of the CYBUG is its ability to seek out its own food source and energize itself. Just as a honeybee is attracted to a flower for its nectar, the CYBUG is drawn to the light of a "sunflower," where it feeds on a meal of raw energy to recharge its onboard battery. Details on how to build a sunflower feeding station are provided in the educational instruction manual.

Although the CYBUG robot has basic survival "instincts" built in, these instincts can be enhanced with the addition of optional daughterboards called HBFs (higher brain function). HBF boards mount on top of the primary CYBUG platform and resemble a small insect head.

HBF 1: Hunger

This add-on board causes the CYBUG to be attracted to a feeding station when the onboard voltage drops below a critical level – in other words, it gets *hungry*. When the CYBUG is fully charged, it becomes photophobic and seeks out the protection of darkness. Materials for construction of a feeding station are included with this add-on kit.

HBF 2: Predacious

The predacious (predator) HBF board modifies the CYBUG's instincts and sensors so that it will track down other CYBUGs and attempt to suck the energy out of them. This add-on kit includes a small transmitter that attaches to the rear of an herbivore CYBUG so that the predator CYBUG can locate its victim.

HBF 3: Programmable

The CYBUG is given a small user-programmable PIC (peripheral interface controller) microprocessor, or "brain." This add-on microprocessor controls the CYBUG's behavior until the power level of the robot drops below a critical level, then the phototropic instinct takes over to find a food source.

The CYBUG is just one member of a new robotic "cyber-ecosystem" that parallels the Darwinian view of nature in a unique and fascinating manner. More than a toy robot, CYBUG and its evolving relatives teach the fundamentals of electronics, robotics, biomechanics and cybernetics.

The CYBUG is suitable for novice to advanced roboticists. Soldering is required. RS&T

resources

For More Information on the CYBUG:

JCM Electronic Services
403.284.2876
CYBUG@nucleus.com
www.nucleus.com/~CYBUG

DISTRIBUTORS

US & CANADA:

Future Active Components
800.655.0006

Mondo-tronics Inc.
4286 Redwood Highway, #226
San Rafael, CA 94903
tel: 800.374.5764
fax: 415.491.4696
info@mondo.com
www.RobotStore.com

EUROPE:

Technology Education Index
www.technologyindex.com

ASIA:

Robotics World
100 Jalan Sultan, #02-16
Sultan Plaza
Singapore 199001
tel: 65.298.8426
fax: 65.298.1372
jstcmpa@singnet.com.sg
www.roboworld.com.sg

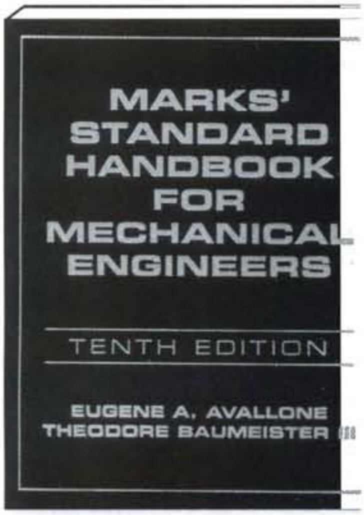
When RS&T correspondents visit competitions, two comments *always* come up: "Wow, there are some really bright people with cool robots!" and, "Man, you wouldn't believe how unprepared some people were."

If you're new to sport robotics, we suggest you get a video of last year's event. Next: Be Brave. Go and Show! Even if you don't win, (or don't complete the course), you'll still have a great time meeting people, sharing knowledge, and learning to make better robots. To help you, here's a video offer from the Fire Fighting Home Robot Contest. This isn't a glitzy PR piece, but it does show, close up, how the contestants ran the course, win or lose.

Send $25 check or money order payable to Trinity College and addressed to: Jake Mendelssohn, 190 Mohegan Dr, West Hartford CT, 06117

Marks' Standard Handbook for Mechanical Engineers

edited by Eugene A. Avallone and Theodore Baumeister III



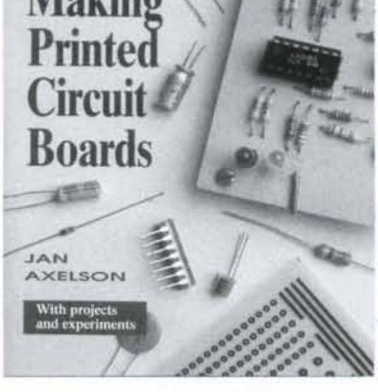
This behemoth book seems to be the "Encyclopaedia Britannica" of mechanical engineering. The 150 contributors to this tenth edition provide us with an authoritative and comprehensive reference tool for engineers and students. If you're a robot builder, or are seriously thinking about designing a robot, we recommend this book for more than light reading. Sections include math tables, measurements, solid and fluid mechanics, heat, strength of materials, properties of common materials, fuels, propulsion, electronic engineering, refrigeration, optics, and more. You don't need to be an M.E. to benefit from this book. In fact, you can learn a lot from the narrative accompanying the tables and formulas.

Recommended sources: McGraw-Hill and other major bookstores
www.booksite.com/mcgraw-hill, tel: 800.352.3566
McGraw-Hill, ISBN 0-07-004997-1, 1656pp hardback, $125

Making Printed Circuit Boards

by Jan Axelson

Hardware adventuress Janette Louise Axelson guides us through PC board fabrication: including drawing a circuit, designing a layout, using CAD software, etching copper patterns, and soldering the components. This easy-to-read, concise but thorough, introduction to making your own circuit boards has some dated (1993) recommendations for software, but the techniques are timeless. Clearly illustrated with many drawings and how-to photos, this book has the most comprehensive soldering section we've seen yet. The index, several extensive lists of resources, and personal tips strategically placed throughout the book make this an outstanding introduction to making your own PC boards.
There are lots of hints and tips that even experienced technicians will learn from.



Recommended source: Most major bookstores.
Tab Books, ISBN 0-8306-3951-9, 327pp pback, $22.9

Booting Up Your Small Robotics Business

It's said that a wise man learns from his mistakes. That's true, but incomplete. So I will now add this: A wiser one learns from the mistakes of others. Fortunately, you've joined a legion of robot builders who picked up this premier issue of ***Robot Science & Technology*** *magazine. You now have the benefit of learning from the mistakes of others.*

When do you need a business license?

Once upon a time, in a neighborhood very near your house, a really smart propeller-head had a Bright Idea (or maybe it was at least a saleable idea).

Being bright and resourceful, PropTop thought about the prospect of Morphing the Bright Idea into Gold, and so sought the counsel of sages in many books, tapes, seminars and mentors.

"Why not get a business license?" PropTop wondered. But the nominal county fee would have to be deducted from the family grocery budget, and filing unfamiliar paperwork at City Hall made PropTop nervous, (although PropTop would never admit this aloud).

> THE SOLUTION TO OVERCOMING THE FEAR OF THE UNFAMILIAR IS TO *JUST DO IT.*

Here's the answer to PropTop's dilemma, with explanations:

Getting a business license means you can begin to establish a *relationship* with a bank. The sooner, the better.

This *will* become important. When you realize you want credit, the banking world will ask, "How long have you been in business?" The *worst* answer is: less than two years. Then, when the bank is deciding how much interest to charge you for your line of credit or loan, they will often give you a better interest rate if you have a relationship established. The same axiom applies when you decide you want to accept credit cards, and then have to deal with merchant card services.

A business license and a *reseller's license* can get you a local sales tax break and has *depreciation* benefits. These may seem unimportant to you if you think that a sole proprietorship is just you, your Bright Idea and your personal credit history. But, in fact, a reseller's license allows you to buy things free of sales tax, if those goods end up in your product to be resold. Otherwise, you'll pay sales tax on the stuff you buy, then pay sales tax again when you sell it as part of your product. You'll want to keep your reseller's permit number in your wallet, because it's as good as cash.

Also, even if you file a simple tax form (with business expenses), you'll discover that depreciation is considered a Good Thing by accountants, bankers and creditors. *You* might think of depreciation as a *loss*, but to creditors, a portion of it is considered as good as actual income.

When you become an official business, your ego will swell temporarily, leading to increased, perhaps premature, spending. (At tax time your accountant will bring your ego back down to size.)

So be sure to read the books about starting your own business, browse the Web for market research and keep banging away in the garage, developing your product. But don't forget to get a business license *before you think you need one.*

General advice from someone who knows: The solution to overcoming the "fear of the unfamiliar" is to *just do it.* First, use your phone book to find a nearby Chamber of Commerce. This "club" of local businesses will happily answer all your freshman questions. Second, call the Small Business Development Center (often attached to a county, state or college agency). Visit them, attend their educational briefings, take notes and always, *always* ask questions, no matter how stupid.

Finally, and perhaps most importantly:

Make a habit of introducing yourself and your dream to everyone, everywhere. You'll be surprised how many folks will relate to you and want to part of your adventure into business.

PS: Use this article's advice at your own risk. RS&T assumes no responsibility for your failures, although we'll be happy to take credit when your business grows.

RS&T

Coming Issues: *Practicing Safe Fax • Intellectual Property Law • What's Hidden in a Business Name When, Why & How to Hire/Fire Employees • True-Life Pitfalls & Pratfalls of Bringing Friends Into a Partnership Thinking Ahead Before Getting a Toll Free Number, SBA Loan or Credit Card • Successful No-Spam Spamming Mailing Lists, the Post Office & Getting Your Money's Worth with Direct Mail What You Should & Should Not Do Without a Lawyer.*

Feedback Loop

RS&T: "Do you think robots will be integrated into home life soon? How will it happen?"
Prof* Ron Arkin, *Georgia Tech: "Sure. Toys!"
RS&T: "Are real robots coming to everyday life soon? How will real robots first insinuate themselves into our daily home life?"
Prof* Rod Brooks, *MIT: "Absolutely. Toys!"

Those Curious Roboticists and Their Flying Machines

You are ahead of your time. There is serious interest in robotics, but only among an exclusive class of people. The rest of society only sees it as a curiosity. *Sojourner* demonstrated to the world the possibilities of robotics, and yet, I only sensed a short-term interest from not only the public but also the media. Once we have a robot do the dishes and put them away, then there will be a serious interest. (I sense this from my wife.)

-Russell A. Buckley

Russell, I'll agree that robotics promises great benefits to the educated classes. But soon "the masses" will benefit. Have you seen the news clips of Eastern Europeans searching for land mines with sandbags on their feet? It's time NOW for robots to be EVERYWHERE. Don't let the media view of the world fool you. They can't report on a technology they don't understand. For a prime example of the limits of the media, see our Pathfinder story on page 36.

Have CanBots, Will Subscribe

OK! I looked over your website after getting the postcard you sent me in the mail, and I liked what I saw. I have two robots (canbots: Richie & No Name) that I've been playing with for some time. I have been looking for just a magazine like this one. I will send in your postcard today. I will try out your new magazine. I hope it's all you say it is.

-Big Rich

Dear Big, thanks for taking a chance on us. I hope you find RS&T useful, informative and entertaining. If not, write immediately with specific suggestions and we'll keep trying to produce a high quality magazine to meet your needs.

There Are No Average Robot Builders (Just Potentially Great Ones)

I am glad that there is finally a magazine on robot construction. I have for the longest time wanted to venture into this field, but there was not much in the field available for the average builder. I would like to get started in the field as soon as possible.

-Dave Johnson

Dave, your letter echoes the sentiments we've been hearing around America during three years of market testing. Stick with us, and we'll construct a whole world of robots.

You Can Never Get Enough Warm Fuzzies

Just a note to say how much I have enjoyed/am enjoying your website. It's obvious a tremendous amount of work has gone into the issue currently online, and as a long-time (but intermittent) robotics hobbyist, I'm grateful for such a resource. I've been interested in robotics since the early '70s, but only this year have I found the information, inspiration and interactivity to really pursue the interest. Your magazine is a great addition to those resources. Thank you for putting your magazine on the Web.

-David Beckham

David, I've waited for a long time to answer your words in print. Finally, the web experiment seems to have paid off. Thanks for the encouraging words. However, we'll keep the website in a slightly different form, from now on: RS&T's OnLine Suppliment. Hope you like it.

WE'D LOVE TO READ YOUR FEEDBACK, WHETHER IT'S POSITIVE OR NEGATIVE.

Send Correspondence to:
Robot Science Technology • 2351 Sunset Blvd #170-253 • Rocklin, CA 95765
or
editor@robotmag.com

MICHAEL A. GREENE

Publisher
Editor-in-Chief

DAVE ALBRIGHT

Art Director
Production Manager

TOM DURKIN

News & Features Editor

KRIS WILLS, M.D.

BioMechanics Advisor

JACOB E. MENDELSSOHN
CONNECTICUT ROBOTICS SOCIETY

The guy who provides the venue for all these robotic fire fighters is Jake Mendelssohn. If there were a Booster's Club Award for the Advancement of Popular Robotics, Jake would win by a landslide. He started his own robotics company in 1975, published the quarterly *Robot Review* in the '80s, and now works with secondary schools, universities, museums, private companies and the film industry. Jake has been encouraging the development of robot competitions throughout North America, and he hosts the annual Fire Fighting Home Robot Contest, introduced on page 4.

ROBIN MURPHY, PH.D.
COLORADO SCHOOL OF MINES

We were pretty impressed by Robin Murphy's leadership at AAAI-97. So it seems natural that she's co-chairing the Mobile Robot Competition at AAAI-98. She tackles tough projects and leads teams with energy and competence. She and her students developed a very cool and useful team of rescue robots, described on page 34.

TERRY KING, COMPUTER ENGINEER
AMERICAN COMMUNICATIONS NETWORK

When we saw our photos of the '97 Fire Fighting Home Robot Contest, one image seemed to capture the whole spirit of RS&T. Terry and his son were hacking a toy tractor as a father-son hobby. Dad eats, sleeps and breathes electronics, mechanics and computers, so it's no surprise that their machine looked so *cool*. Terry seems to be a pretty good photographer and writer, as you'll see on page 5.

"REPORTS OF THE PROBLEM IN THE POPULAR PRESS WERE GENERALLY INCOHERENT, AND IN SOME CASES WILDLY WRONG..." *see Pathfinder page 36.*

John Piccirillo, Ph.D.
Sigmatech

Sometimes the most useful robotic applications come from multidisciplinary thought: John Picccirillo has degrees in physics and astronomy but spends his days working for a defense contractor, his nights on robotics, and his weekends on a house. An amateur's amateur, his robotic interests include decentralized control, case-based reasoning, and anything else but vision systems. John has graciously assisted RS&T for three years, providing sound guidance, useful articles, and supportive friendship. He also developed a two-time award-winning autonomous fire fighter, detailed on page 9.

Ronni Katz, veteran Robot Warrior
Quest Inc.

Well, it's nice to know about all that circuitry, but while the RS&T staff was touring competitions last year, we noticed that a whole lot of people really don't know beans about metals and plastics. So we approached Ronni Katz to help fill us in on aluminum. She managed to take time out from her video production company (and her new novel) to interview her favorite metal benders for us. Thanks Ronni, for giving us newcomers a great introduction to working with aluminum, starting on page 46.

Karl Lunt, Seattle Robotics Society

After reading about these autonomous machines, a lot of our readers will want to learn their first lesson about microcontrollers. We found the perfect instructor in veteran writer/robot builder Karl Lunt, who has written nearly a hundred articles on robotics. But no editor has ever met him, and his picture has never appeared in public, so we can't really be sure that he exists. Karl is obviously an outstanding programmer and amateur robot builder. In fact, he's the only person who never had to learn SBasic. His teachings begin on page 40.

"WHEN MOST OF THE PIECES WERE WORKING, I DECIDED IT WAS TOO UGLY, SO I RE-BUILT IT COMPLETELY..."

see Fire Science Meets Robot Technology page 9

Random Firings

Going Where No One Has Gone Before...

Three years ago, we asked, "How could we bring robotics to the world the way the Techno-Nerds of the '70s brought PCs into our homes?" The challenge was to provide information that experienced robot builders could use, and that newcomers could learn from. You are, we hope, holding the answer in your hands.

There were a few dozen books that general audiences could read, and tens of thousands of web sites. But there were no magazines that informed and entertained the neural wet nets in most of our brains. Other similar publications attempted to survive on smaller audiences, with less advertising, in a tighter economy, when robots were not nearly as advanced, and when the population was not nearly as technically literate. Some journals were available, but they were really expensive and way above the reading level of most folks who would like to read about robots.

Roboticists express their skills in many forms: artificial intelligence (AI) software, mechanical creatures with various controllers (autonomous, firmware, teleoperation, radio control, brains on- and off-board, etc.) and in simulations. They are found in hundreds of schools, dozens of clubs and in commercial start-ups spawned by grad students. Roboticists vary in education: hardware types, software types, wannabees, PhDs and thousands of "amateurs" (like Einstein was an amateur?) Those are the audiences we are trying to join together. Quite a task, but then, robotic builders tend to share their inventive ideas easily. We're quite a gregarious community!

Call me crazy (many have), but it's my belief that our community of readers (educated and talented) will support this magazine with our collective pioneering spirit. I'd love to hear your thoughts on this, both positive and negative.

So, welcome to a robot magazine aimed at the Whole Planet! —Mike

How About that Anti-Asteroid Task Force?

You and I may know that a space-based intelligent nuke delivery robot could save the world, but when Assistant Secretary of Defense Kenneth Bacon was asked recently about an "Anti-Asteroid Task Force," his response, "We're not very far along," was not much of an answer.

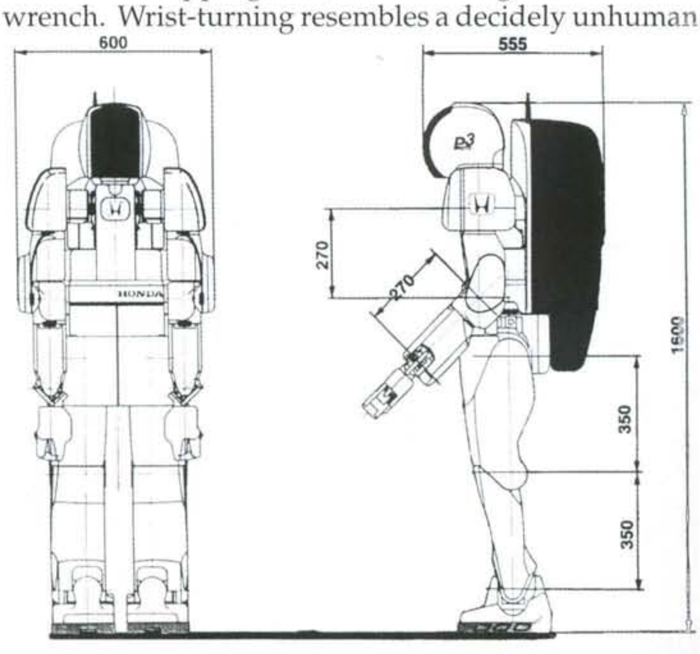
A Pentagon spokesperson informed RS&T that although Under Secretary of Defense for Acquisitions Paul Cominsky suggested the idea after a TV mini-series destroyed much of L.A. last year, the task force never actually met. In fact, no one was even appointed to be on the committee, and there are no meetings scheduled in the future. "We were overcome by higher priorities," the spokesperson said. "If this were a pressing problem, it would be global."

What's Honda Up To?

"This robot is so secret, don't even bother calling to ask questions," we were warned. Indeed, when we called Honda in Japan, they hung up the phone as soon as we mentioned the *P3*. Later calls yielded some vital measurements and a couple of photos. Our source told us that Honda will continue to develop a line of autonomous humanoid robots.

Honda's latest humanoid robot, *P3*, stands 1600 cm (5'5") and weighs 130 kilos (286 lbs).

This is a significant improvement over her predecessor, the *P2*. We've seen tapes of *P2* in action. She walked with a slow purposeful shuffle at 2 km/h, and climbed stairs with a graceful gait. She demonstrated walking, turning, ascending and descending stairs, pushing a standard shopping cart, and turning a nut with a wrench. Wrist-turning resembles a decidely unhuman



triple-jointed motion. Figure the Jacobians here!

This beast's beauty is in her weight-shifting behavior: a slow-motion wiggle that reveals the complexity of her self-balancing mechanisms. She can balance upright on a slope, or if she is pushed. Her legs have two joints with 6 Degrees of Freedom, arms have two joints with 4 DOF, and each hand has 2 DOF. And those are force-feedback sensitive grippers.

P3, the new incarnation, is much slimmer, closely resembling a human in a spacesuit. *P3's* AC servo motors are powered for 25 minutes by a 136V 6Ah nickel-zinc battery backpack.

Two Micro SPARC II CPUs run at 110 MHz to coordinate *P3's* gyros, G-sensors, cameras and 6-axis force sensors on her wrists and feet. When desired, a wireless ethernet interface connects her to a human operator's platform for radio control.

I wonder if Kazuo Hirai knew he was designing this month's centerfold?

PCs Are Passe'

If a robot is a computer with arms and legs (or wheels, or not), then you might expect a more powerful processor would destroy any competitor with fewer MIPS (millions of instructions per second).

Not so. In fact, when two autonomous machines faced off at Robot Wars last August, Bob Gross' robot, sporting a Parallax Basic Stamp II, flipped a 386-powered chain-saw-wielding Camp Peavy production in the first few seconds of the fight.

Marc Thorpe's annual Robot Wars® in San Francisco is one of the most exciting places on Earth for three days, and you'll see us there, digging into the programming of the autonomous entries, and reporting on the engineering of the radio-controlled walkers. Sure, it's the "Monster Truck Rally of Robotics." But some of science's best minds are twisted, too.

Are We LIS?

Not really. But let's face it, *Lost in Space* captured the imaginations of a generation, and probably encouraged some of us to pursue robotics as a career. RS&T is non-fiction, but we thought you'd like to know something about the various robots that appeared in the TV series 1965-68 and in the new movie that premieres this month. Check out our interview on page 64. RS&T

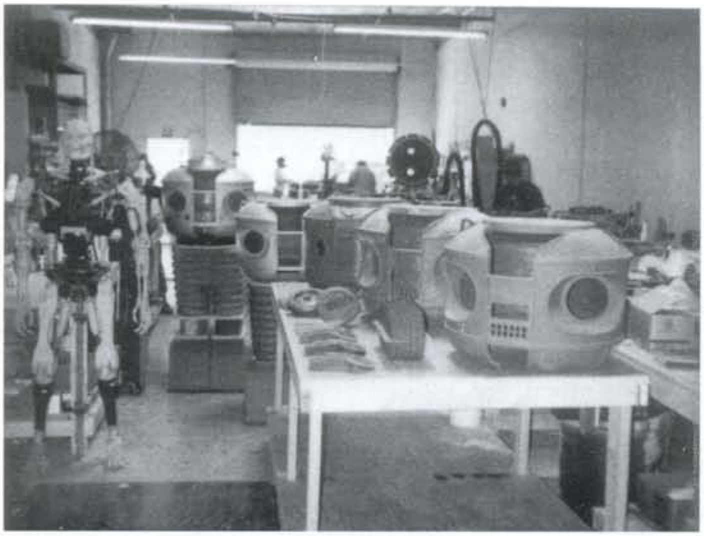


*Photo Obtained by RS&T Intelligence Agents
Shows the ICONS Assembly Line
Preparing to Dominate the Earth with Robots.*

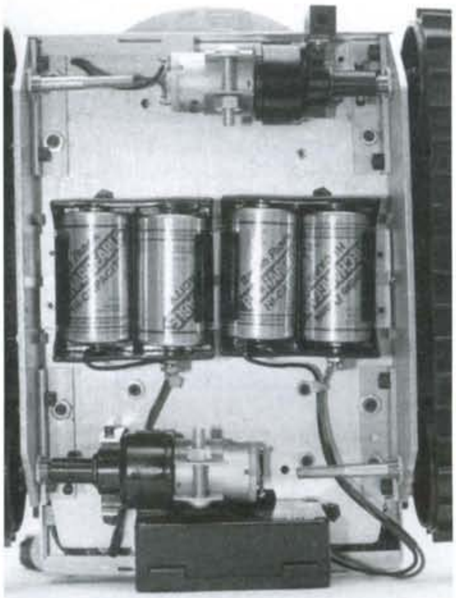
F.E.A.R. continued from page 7

be synchronized in order to advance, reverse and turn in a coordinated fashion. Some designs have employed stepper motors, while others use servos. Some people use experimental nitinol shape memory alloy (SMA), such as Flexinol Muscle Wire from Mondo-tronics (www.robotstore.com). This unique material can be stretched and contracted by regulating the flow of electrical current.

For our fire fighting design, we chose to use tracks. After searching from store to store, we settled on a Tonka Toy bulldozer. After performing a mechanical hack, we attached the tracks, track wheels and side frames to an aluminum base.

What Materials to Use?

Plastic? Aluminum? Wood? Many hobby kits are plastic, and some come with a sheet-metal base. Legos are popular with students, because there is little or no cutting or drilling involved. Legos are also relatively lightweight. Aluminum is the choice of more advanced or heavy-duty designs. It requires machining, however.



FEAR was driven by two cordless screwdriver motors (top and bottom of photo). The motor was powered by four rechargeable batteries anchored to the undercarriage.

Your decision as to what material to use to build the robot will be, of course, greatly influenced by what the robot will be doing. If its only purpose is to propel itself around, then a minimum lightweight design is appropriate. If it has to transport heavy batteries or cargo, then a more sturdy construction is called for. Lifting, pushing or pulling will also require stout construction.

It is surprising how far a little imagination will go when trying to find parts for your project. For example, in our robot, a transistor radio antenna made an excellent boom for the water sprayer. In fact, the sprayer itself was hacked from a hand-held squirt bottle.

Avoid Tim-the-Tool-Man's Mentality

Keep the design as light as possible. Heavier construction requires more powerful motors. More powerful motors require bigger batteries. This eventually becomes a vicious cycle. Speaking from personal experience, this is an easy trap to fall into.

Also, a loaded-down motor draws extra current. Excess metal, heavy batteries or other weighty features all contribute to overloading the drive system. A poor choice of gears can also prevent the motor from operating within its normal range.

For our robot, we found some gears to raise the torque and bring down the rpm. The problem with this approach was having to devise a way to mount the gears and keep them precisely aligned. Ultimately, in our search for parts, we ran across surplus DC (direct current) motors supplied by Mendelsson Electronics (www.meci.com), which were originally designed for cordless screwdrivers. They were complete with planetary gears for good torque. The current draw is a little higher than desired, but otherwise the motors were perfect. We found that 1-inch electrical conduit clamps worked nicely as motor clamps.

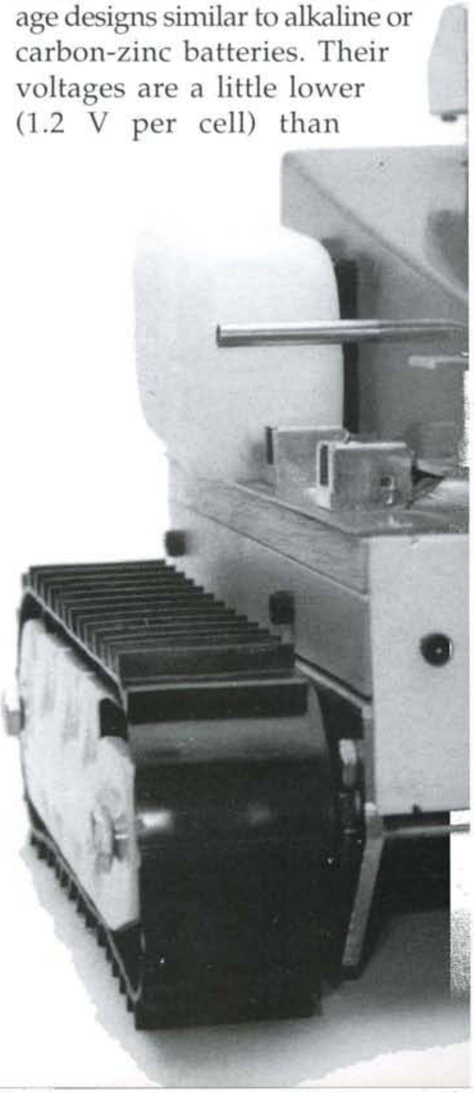
Which Batteries Are Best?

Common battery types range from nickel-cadmium (NiCd), gel cells, lead-acid and alkaline.

Alkalines are generally used in lightweight applications. The current draw should be limited, as they are not rechargeable.

Gel cells and lead-acid batteries come in a good selection of current and voltage combinations. Their size can vary some, but they usually take a lot of room and tend to be heavy.

NiCds are popular because they can supply awesome current in package designs similar to alkaline or carbon-zinc batteries. Their voltages are a little lower (1.2 V per cell) than



standard cells, but this usually is not a problem.

To ensure that NiCd batteries don't degrade over time , they must be fully discharged before being recharged. This assures that they will continue to output their full potential and remain in good health. Some of the newer rechargeable batteries (e.g., Rayovac Renewals), however, are just the opposite – their long life and capacity depend on *not* being fully depleted.

What Sensors to Use?

Sensors can be a complete study by themselves. (See *Sensors for Mobile Robots Theory and Application* by H.R. Everett.) Your selection of sensors depends on several things. What are you trying to accomplish? What kind of environment will your robot be operating in? What variables could interfere with the operation of the sensors? Is your robot operating autonomously or as a dead-reckoner?

Robots in the fire fighting contest certainly need sensors to detect the a flame. Some may also use bump switches to detect an obstacle or wall. Some may use infrared or sonar for object avoidance. Most will employ some type of shaft encoder to sense or track distance traveled.

Certain sensors can be drastically affected by the robot's own vibration. This can be a problem with tall track-style drives. Electric currents and permanent magnets in motors can distort the readings of electronic compasses as well.

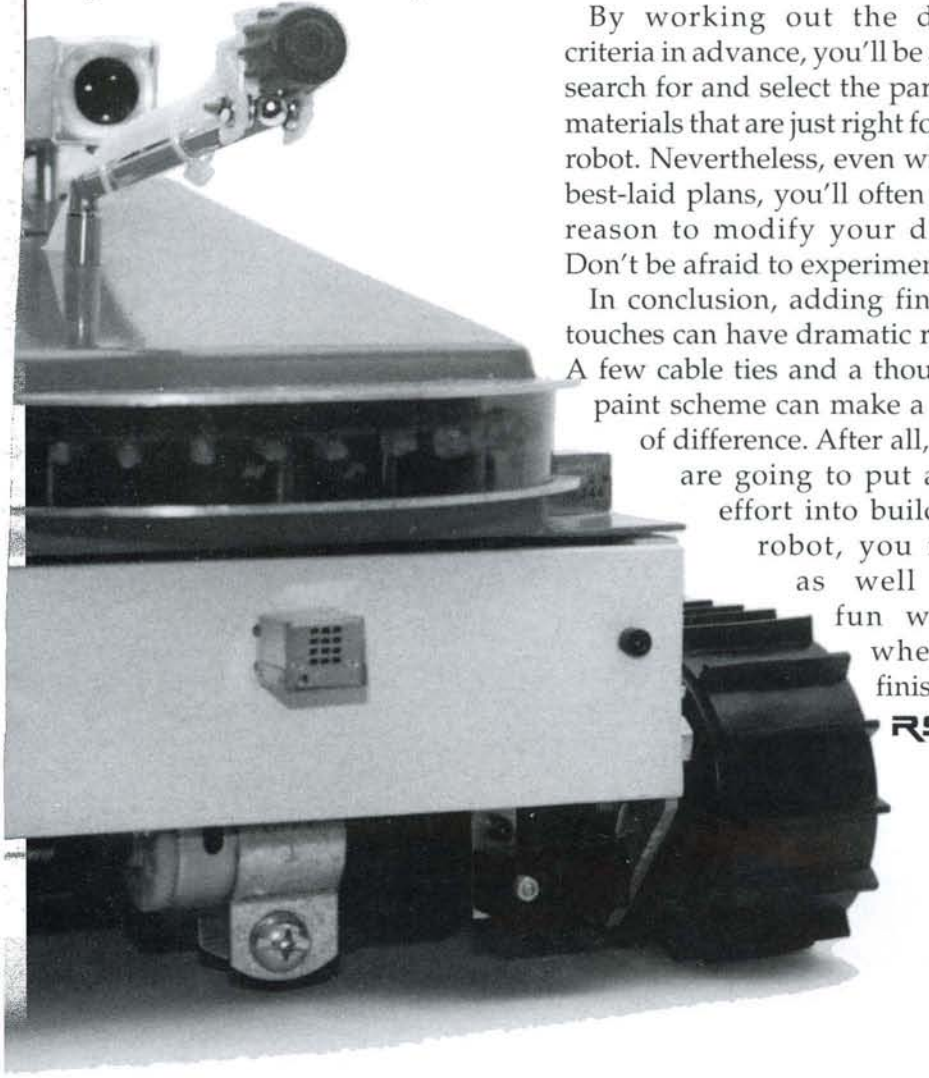
Other sensors include bend sensors, photocells, thermistors, EMF sensors, etc. Some sophisticated designs are even employing rudimentary vision.

Build a Robot for Fun!

By working out the design criteria in advance, you'll be able to search for and select the parts and materials that are just right for your robot. Nevertheless, even with the best-laid plans, you'll often find a reason to modify your design. Don't be afraid to experiment.

In conclusion, adding finishing touches can have dramatic results. A few cable ties and a thoughtful paint scheme can make a world of difference. After all, if you are going to put all the effort into building a robot, you might as well have fun with it when it's finished!

RS&T



Author's Recommendations

Images Company
P.O. Box 140742
Staten Island, NY 10314
tel: 718.698.8305
fax: 718.982.6145
www.imagesco.com
Specializes in items for robot experimenters, such as voice-recognition boards, digital and analog compasses, tilt switches, bend sensors and more. Be sure to ask for their catalog.

The Electronic Goldmine
P.O. Box 5408
Scottsdale, AZ 85261
tel: 800.445.0697 (orders)
fax: 602.451.9495
www.goldmine-elec.com
Surplus electronic parts and kits. Features online catalog.

MECI (Mendelsson Electronics Co.)
340 E. First St.
Dayton, OH 45402
tel: 800.344.4465
meci@meci.com
www.meci.com
Vast surplus electronic, mechanical stock.
Motors, LEDs, controls.
Online catalog.

Arrick Robotics
2107 W. Euless Blvd.
Euless, Texas 76040
tel: 817.571.4528
fax: 817.571.2317
arrick@ix.netcom.com
Arrick makes a number of automation systems, including stepper motor systems, positioning tables and robotic workcells.

Additional sources on the World Wide Web:
www.parallaxinc.com.
Parallax Inc., BASIC Stamp supplier

www.gleasonresearch.com
Handy Board supplier

www.robotstore.com
Robotic books, motors, gears, Flexinol Muscle Wire, batteries, etc.

www.alltronics.com
Electronics, Parallax products

INTERVIEW: VERNER GRESTY

Robot Builder for *Lost in Space*

"Building robots is fun, it's really quite a thrill."


Photo courtesy Jim Henson's Creature Shop (thanks, Fanny)

When our publisher suggested we should follow *Lost in Space*, an editor made the point that we are definitely *not* a sci-fi magazine. But hey, the campy 1965 TV series was *inspirational* to a lot of us, and frankly we couldn't ignore it this month. So we called Verner Gresty, the Head Guy in Charge of Building Robots for *Lost in Space*.

Usually, Jim Henson's Creature Shop in London tends to make small and light animatronic puppets. But this time, the hardware was, in itself, a challenge that all robot builders could learn from.

Robot #1, which you'll call *Big Blue* when you watch the movie, weighed 3000 lbs., and could lift a grown man easily. In fact, in one scene, character Don West actually mounts the robot.

Onboard electrical motors wouldn't have had the power density demanded by the role. So they bought hydraulic rams off the shelf, and powered the robot with a TexTron 20,000 psi hydraulic generator. Yup, the robot is tethered, and can still make 15 mph, flat out.

Did they use real sensors on the robot, to get the bot's point of view? Oh, no, no way. This wouldn't give enough perspective to safely drive the robot at high speed, so close to the actors on the set.

So four crew members controlled the machine, communicating with Verner through headsets. Two of the crew wore light-weight virtual reality suits. One controlled the arms, while another controlled the torso. Two others controlled the head and tracks with joysticks. For added safety, the man who controlled the locomotion was positioned directly in the "sight line" so the robot was always driving directly toward him.

"We wouldn't trust radio control with a creature this powerful on the set," Verner told RS&T.

There were also four levels of safety in the motion control software. And Verner carried a powerful radio transmitter connected to a robot-mounted "kill" switch. Fortunately, he never had to use it.

All of the control signals on 60 channels were electronically recorded for instant playback, so that if the director liked a particular sequence of motions, all 60 channels could be duplicated perfectly for the next take.

So, what's the process they used to design the bots? Well, from conceptual images, they made 600 AutoCAD® drawings to help make three different sized models. Then they chose to go with the medium-sized monster, using fiberglass and traditional molding to effect the sharp corners they wanted. They used aluminum for the chassis. Engineers worked for 14 weeks to build two full-sized copies, making *no* lightweight models for special effects.

And in case you're wondering, Robot #2 is actually Robot #1 that has been cut down, just like young Will Robinson does in the movie.

Robot #2 looks innocent enough from the front, but it retains a giant and powerful scorpion arm on his back, so that he can be quite menacing while guarding the Robinson family.

So, why didn't the new *LIS* robot look more like the old one we remember from television?

"Well, this is a far more futuristic robot than was used in the TV series, which was obviously a man in a suit," Verner said. "*Blade Runner* set the scene for robots of the future. Movies like *Dune* and even *Starship Troopers* show how society will cope with technology."

And considering the progress we've noted in artificial intelligence in recent years, we at RS&T suspect that our children, or at least our grandchildren, may be living with the likes of the *LIS* robot. "It's going to be wonderful when robots finally integrate with families," Verner said.

OK. Now you know more about the world's most famous robots than any other kid on your block. So take a look at our really neat pictures on page 62.

RS&T

Continued from page10

If a candle wasn't found, Marv would go to the next doorway. This was the grand plan.

I also wanted to replace the water sprayer but I wasn't sure what else to use. I didn't like fans because they required getting close to the candle, which would have taken more time.

Getting Started

The first step is putting together the platform deck, motors, and wheels. Since the salient feature of a mobile robot is mobility, I like to start with the locomotion, because this sizes the rest of the robot.

This phase took me about 3 to 4 aggravating months. I first tried the motor/gearbox I had used for Marv-94, but the motion was too sloppy. Then I tried different DC gearhead motors, but even with better wheel encoders the turns were still inaccurate. The problem seemed to be that after the motor was turned off, the platform coasted. I tried braking and briefly reversing the motor direction but that wasn't good enough.

There was also the problem of getting the motors to run at precisely the same speed. Even though DC motors can be made to dead reckon fairly accurately with a good control algorithm, I decided to try stepper motors. It took another month to find wheels that I could adapt to fit on the stepper motor drive shaft of the size I was looking for. An incredible three months went by before the platform could move soberly.

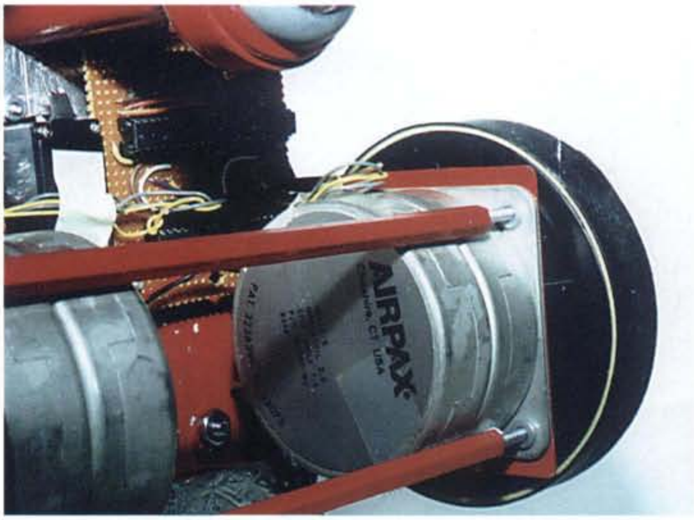
Motor and Wheels

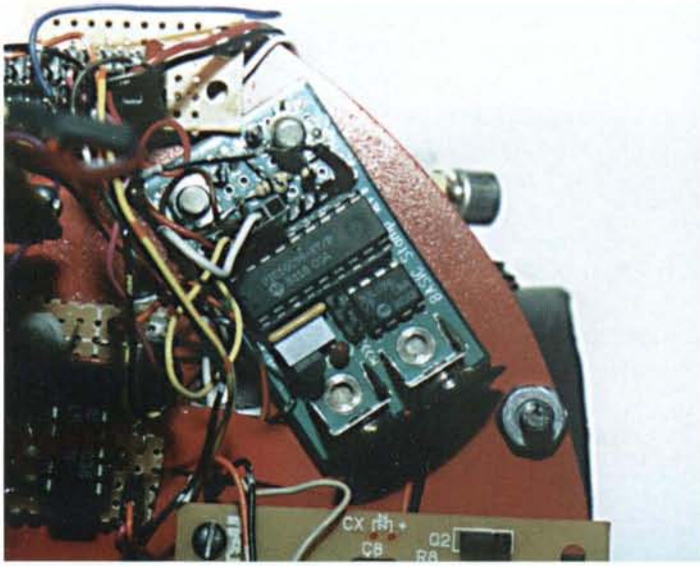
The basic platform is an 8″ diameter disk cut from finished 3/8″ plywood. Two stepper motors attach to a common U-shaped bracket, which is bolted to the plywood disk. The motors are unipolar steppers. They operate at 5 V and 1.5 amps/phase, for a holding torque of 36 oz-inches, which is barely adequate (I'll discuss the stepper motor control later). Two long spacers connect the open ends of the U for rigidity.

The wheels are hard plastic, 4″ in diameter and 1″ wide. They came with a center bearing, which I knocked out.

Whenever I buy a new part, I buy extras. I bought four wheels and wrecked two trying to remove the bearing.

The center hole was too big to fit on the quarter-inch motor shaft. I drilled it out wider to 1/2″ and attached the motor shaft with a coupler from Small Parts Inc. This had a 1/2″ outside diameter and a 1/4″ hex hole on the inside. I attached it to the motor shaft flat by drilling and tapping the coupler for a 4-40 set screw. I then drilled a hole for a 6-32 screw radially





Top: Marv's underbelly. Note left and right stepper motors and traction tape around the tire.
Bottom: A Parallax Board with Basic Stamp Rev-D.

through the hub of the wheel and into the coupler.

This wheel isn't going to slip on the axis.

Next, I improved the traction by covering the wheel rim with a layer of double-sided foam tape topped with rubber electrical tape.

The drive wheels are mounted in the center of the platform so that the robot can turn about its vertical axis. Marv also has a third, idler wheel. This is *not* a caster. Casters (the crude ones that are commonly available), impart a sideways push when they turn. Instead I used a 1″ roller ball - a ball mounted in a universal joint so that it can roll in any direction. There's no horizontal axis.

Computer

I suppose this part could have been quite involved, but I chose to make it simple.

Use separate power sources for motors and electonics to prevent motor noise from affecting processors.

I determined that I didn't need high speed, large memory, or many other amenities that are available. I chose the Basic Stamp II (BS 2) from Parallax (www.parallaxinc.com). It uses a form of interpretive basic, has enough EEPROM memory (2K) for about 600 lines of code, and is slow (An empty For/Next loop takes a half millisecond to execute.). It is also small (1 1/4" by 5/8"), consumes little power, and has 16 bi-directional I/O lines.

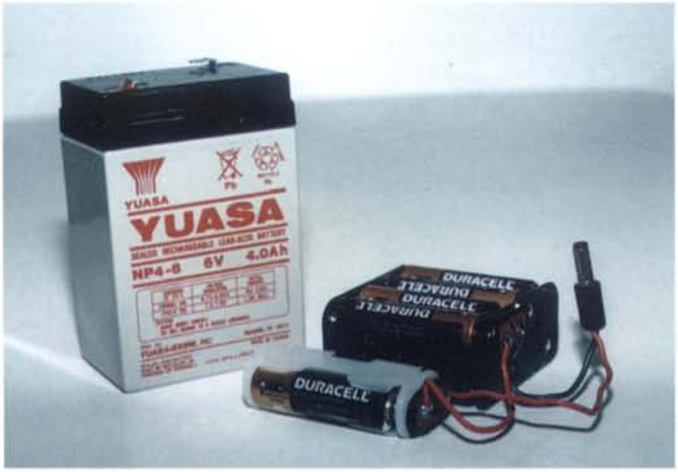
I also used one Basic Stamp I (8 I/O lines, 256 byte memory) for control of one of the servos.

The development environment is very easy. Code is written in a simple editor and downloaded from the PC to the Stamp with one command. A Debug command is available for returning variable values back to the PC.

The completed robot used 15 of the 16 I/O lines and about 90% of the memory.

Battery Power

Marv has two supplies: a 6 V, 4 Ah gel cell battery for the stepper motors and two servos, and a 12 V NiCad battery pack (10 AA cells) for the electronics. The grounds are common. The batteries rested on the two, 6" long spacers that braced the motor bracket, a snug fit under the platform. Some of the electronics needed 12 V, some 5 V. A 7805 regulator is used to provide 5 V. The motor and electronic supplies are switched separately. Since steppers draw maximum current when they're not stepping, they can remain off when not needed for testing. This is helpful during program downloading and sensor testing.



Stepper Control

There are many different kinds of stepper motors and control options. A good reference is the stepper FAQ at www.cs.uiowa.edu/~jones/step. Marv's steppers are unipolar, 4 windings. After much searching, I used a SGS-Thomson stepper controller chip, L297, and a 2 A quad Darlington switch, L702B. The controller uses inputs such as reset, direction, step clock, half/full step to generate the appropriate sequence to the stepper windings. This could be done by the controller but I prefer the dedicated hardware. The Darlington switch is required to amplify the controller output to the current necessary to control the motors.

The L297 controller has additional functions that I did not use. (In retrospect, this was probably a mistake, since Marv could have been made speedier.)

Stepper control comes in different modes and drive techniques. Modes include full, half, wave, and micro stepping. These modes allow the basic step size to be subdivided, but at the cost of reducing the torque. Marv uses a full step size of 7.5 degrees/step.

Drive techniques include direct, L/R, and chopper. Direct drive uses the rated voltage to drive current through the windings. An L/R drive uses a higher than rated voltage to force current more quickly through the windings and a series resistor to limit the max value. A lot of energy is wasted in the resistor but the stepping is faster.

A chopper drive is the best. It also uses an overvoltage, which it monitors with a small sense resistor in series with the motor windings. When the max current is reached, the controller switches off the current. This repeats at the chopper rate frequency, usually many kHz.

Navigation Testing

Enough robot is now at hand so that the platform can move autonomously and attempt to negotiate the maze. Actually, this is a never-ending process. As other robot functions are added, new navigation complications arise.

I began by learning how to work the motors. Steppers are low torque compared to comparable DC motors and the torque decreases as the motor is stepped faster. Therefore, the motors must be accelerated to their final step rate or they will miss steps or stall completely.

At this point Marv weighed about 7 lbs (mostly motors and batteries). By experimenting, I found an acceleration cycle that brought Marv to its max speed without missing steps. A shorter deceleration cycle brought Marv to a stop. The limiting speed performance is a function of the basic motor capabilities, Marv's weight, and the direct drive circuit.

To finish the navigating task, I had to pick a pre-planned route.

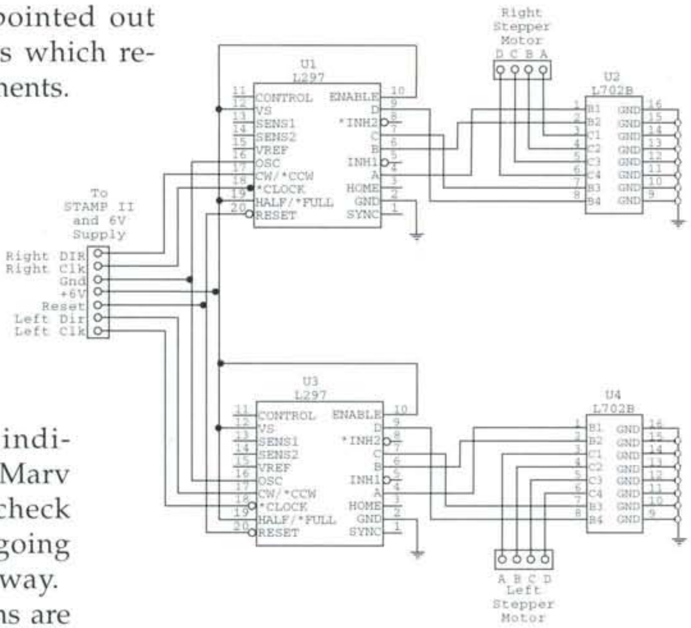
I had run Marv through the maze so many times I could eyeball a one degree misalignment.

Looking at the house diagram, I decided that by turning left at the end of the first hall it was possible to search three rooms with no additional turns - a time and dead reckoning advantage.

Marv has two UV sensors, one on the left side, above the wheel, and one on the right. Marv first stops in front of room B and counts the UV pulses received for two seconds. If no candle is indicated, Marv backs up to room A and samples it. Next Marv backs up to the other end of the hall and samples room C. If this is also negative, Marv moves forward to the last unexplored hallway, turns right, moves to the doorway of room D and searches it.

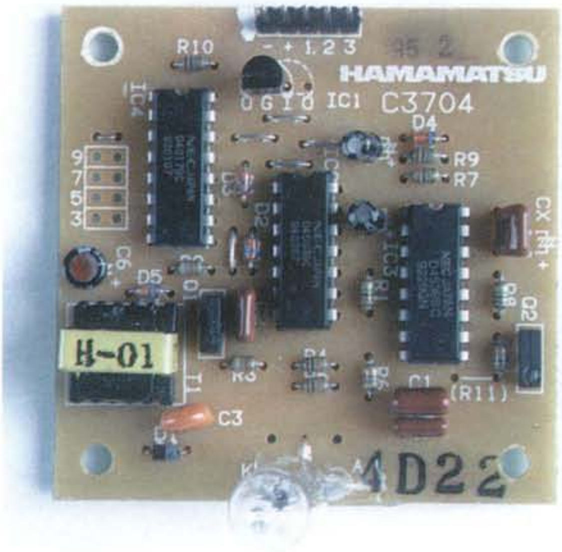
This was Marv's basic search pattern. Testing pointed out some failure modes which required some refinements.

Depending on the candle location in room C, the UV sensor will occasionally miss it. Since this is a rare occurrence, the basic search pattern is still used, but if the candle is not indicated after room D, Marv moves back to re-check room C; this time going to the room C doorway.



All of Marv's turns are in-place pivots except the first turn at the end of the first hallway. That turn is made by pivoting around the left wheel to double the angle resolution of the turn.

The number of steps required to execute each path segment of the basic search pattern is calculated with the known step size and wheel diameter. I refined these numbers by trial and error. Eventually, Marv could move through the search pattern flawlessly. That is, stopping at each candle search position without bumping into the walls. One source of error was Marv's initial orientation when placed in the maze home circle. Although the rules allow the use of an alignment fixture, I found it unnecessary. I had run Marv through the maze so many times I could eyeball a one degree misalignment.



The phototube protrudes upward at bottom of the board

Detecting The Flame

Marv uses two flame detectors. A UV flame detector for the candle search mode and a PIR detector for pinpointing the candle location inside a room.

—UV Flame Detector

The UV flame detector is a professional unit from Hamamatsu. It consists of a UVtron phototube (R2868, about $40) and a UVtron driving circuit (C3704, about $35). Marv used two of these, right and left, in order to search rooms without turning.

The UVtron tube is soldered directly to the driver board. The board has a changeable jumper for setting the background threshold to 3, 5, 7 or 9 counts, positions marked on the board. The board puts out a 10 ms pulse only after the set background count is reached within two seconds. I wanted as sensitive a detector as I could get, so I set the jumper to the unmarked position just past the 9 point. This produced a pulse out for every UV photon detected. (I preferred setting the background discrimination level in software.) This was convenient for making adjustments during the testing day before the contest.

The output of the UV detector depends on the position of the candle in a room. If the candle is a few inches from the detector, as it would be if it were in the doorway, the detector produces about 20 counts/sec. (This is the saturation rate.) If the candle is in the room's hidden corner, not directly visible from the doorway, the measurement is about 4 counts/sec.

—Pyroelectric Detector

Once the UV detector discovers a candle in a room, Marv pivots to face the room and searches for the candle position with the PIR (passive infrared) detector. The PIR uses the pyroelectric effect of a Lithium Tantalate crystal to convert radiant energy in the mid-infrared, 8 to 14 microns, to an electrical charge. The detector package uses two elements to detect a differential effect, making it a good motion detector.

In this case, the motion is due to scanning the detector across the room with a servo. The particular PIR unit I used is an Eltec Model 442, which has an internal voltage regulator, reference, and amplifier in a TO-5 style windowed package. In order to increase the sensitivity of the detector and restrict its FOV, I used a small IR-transparent Fresnel lens, held in front of the detector with a light cardboard cone. The detector package was bought as a kit from Acroname (stever@acroname.com) for $45.

There were several problems that I had to overcome before the PIR detector would be suitable for the contest. The output needed amplification and discrimination.

The PIR output from the detector rides on a 2.5 V level. It can be a positive or negative pulse depending on the scan direction. The amplitude of the pulse depends on the heating of the detector elements, thus it is sensitive to the scan rate. For a candle flame just 4' away and using a reasonable scan rate, the pulse is only a few tenths of a volt from the DC level.

> I WAS TESTING IN MY GARAGE IN THE WINTER. IT TOOK ME A LONG TIME TO DISCOVER THAT THE PIR WAS TRIGGERING ON WARM AIR CURRENTS

The signal also needed amplification to detect a candle at the limits of the required FOV. I positioned the PIR detector to be at the same height as the average flame, 7." (The rules stipulate the candle will be from 6" to 8" tall, and a candle flame is about an inch high.) For the range of candle heights and distances (a few inches to 4') the PIR must have a detection FOV of +/- 10 deg. To interface with the microcontroller, the PIR output needs to be a simple 5 V pulse. After amplifying the signal, it is passed through a window comparator. This compares the amplified signal to a threshold (0.6 V in this case) and pulls it up to 5 V.

Testing the PIR with candles at various distances and heights, I set the amplifier for a gain of 9. Amplifying too much can produce spurious detections.

Be aware that not all candle flames are the same. A newly lit candle gives off less than half the radiation of a candle that has been burning for a few minutes.

The PIR detector with Fresnel lens is attached directly to a piece of circuit board (about 1 1/2" by 3"), which contains the amplifier and comparator circuit. This unit is attached to the horn of small servo motor, Futaba S-148, with a small aluminum bracket.

At the beginning of a scan, the microcontroller positions the PIR to a CCW (counter-clock-wise) start position, then slowly scans it CW.

The beginning and end scan positions are chosen separately for each candle scan position. Since the scanning is a little slow, the positions are chosen to cover the room without wasted motion (see the floorplan).

The first scan is made outside a room in case the candle is just inside the doorway. If the candle is found, the platform is oriented to point to it and the CO_2 extinguisher is discharged. If the candle is not found, Marv enters the room

Note that the cone with Fresnel lens is protruding from the board.

several inches and scans again to look in the hidden corner.

Given that the UV detector has indicated the candle is in the room, a negative result for the PIR scan is taken to indicate that the candle is hidden behind the furniture. In this case, Marv would advance further into the room and scan from a new perspective. In practice this scheme works for all possible arrangements of candle and furniture.

Except one. During testing, I became paranoid about bumping into the simulated "furniture" while advancing into a room looking for a hidden candle. So in the last month before the contest, I added a three-part bumper to parry this threat.

The bumper uses microswitches to sense a collision. If Marv runs into the furniture, he stops, backs up and discharges the CO_2, which is powerful enough to flow around the furniture and snuff out the candle.

During testing in the maze, I experienced a lot of problems with the PIR detector. By now it was about November and I was testing in my garage. In order to take the chill out of the air I installed a heating unit with a blower. It took me a long time to discover that the PIR was triggering on warm air currents! At first I thought that maybe the detector was seeing beyond the maze walls and detecting something outside.

Programming the microcontroller to scan back and forth repeatedly showed that it did not respond to IR sources, such as a hand, placed outside the maze. Mystified, I programmed the detector to scan, stop when a source was found, note its position, and re-scan. If the source was found in the same place twice, I counted it as the candle. If not, it continued to scan. This was time consuming and worrisome. After wasting several weeks, I finally realized what the problem was and eliminated all the unnecessary code. I also discovered later that the PIR would detect a warm place on the wall *where a previous candle had been*, or detect a hot, unlit candle. The lesson here is that extensive testing is necessary to find the failure modes.

Another "feature" of the PIR discovered during testing is that it can take 20 to 40 seconds for a "cold" unit to begin working when first turned on, Then the PIR emits a transient pulse. A small grain of wheat light is turned on with the electrical power and the PIR output is monitored for the initial pulse. When this occurs, the light turns off, signaling that Marv is ready to begin.

The Flame Extinguisher

As previously mentioned, I used a CO_2 extinguisher. This worked much better than expected. A two second blast blows out a row of candles two feet long at a distance of four feet!

I was fortunate to find a hefty CO_2 cylinder that came with a handle-operated control valve. The extinguisher (with valve) is 7" long and 1 1/4" in diameter. The cylinder holds 38 grams of CO_2, enough gas for 10 seconds of operation. The cylinder was originally sold as a duster and is available from Hosfelt Electronics (800.524.6464) for $15 (cylinder and valve) or $5.50 for a replacement cylinder.

The extinguisher is mounted on the front of the platform and is operated by a servo which pulls down on a metal lever attached to the control valve handle.

The extinguisher serves an additional function. The cylinder bottom is hemispherical and rides about 3/32" above the floor, thus providing a fourth support point. This is not used when Marv accelerates or moves at constant speed but comes into play during deceleration, or when Marv accelerates going backwards.

Sound Activation

Since Marv was going whole hog on the bonus opportunities, I added sound activation as well. This also turned out to be handy for restart in testing modes. The stipulated buzzer is a Radio Shack #273-075 that emits a 3.5 kHz tone. I put this in a little project box with a 9 V battery and a momentary push-button. The detection circuit used a microphone with a built in FET amplifier (Radio Shack #270-092), a 741 op-amp, and a 567 tone decoder chip. Holding the buzzer close to the microphone and giving a one second blast suffices. I also incorporated a push button override in case the sound activation failed.

Putting It Together

Many parts were tested in parallel. The UV detector and the extinguisher were first tested alone while trying to find a workable motor configuration. The PIR detector was added to the platform later.

I made many changes during this

The CO_2 trigger (top) is connected by a ball-capped link to a servo.

process. At one point the wheels were mounted in the back of the platform and the roller ball in the front. The motor bracket was changed at least three times. The PIR was first mounted near the front, off to one side. Wires went all over the place.

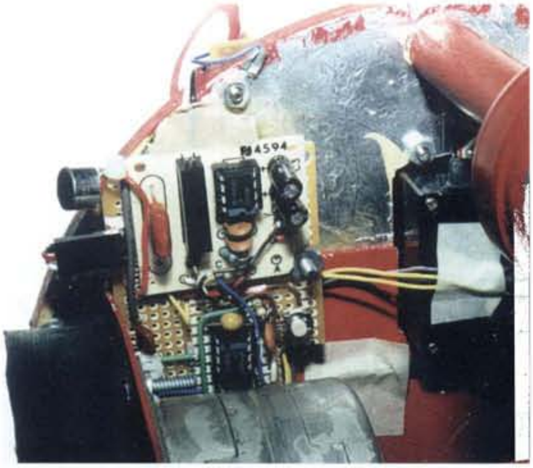
There was a second platform above the main deck that held the computers and some electronics. When most of the pieces were working, I decided it was too ugly, hard to work with, and likely to have wires pulled out, so I re-built it completely. It was during this stage that a simple layout came about.

Typical Operation

Let's go over a typical run (refer to the floor plan). A small toggle switch turns on the electrical power and a red LED. The motor power is turned on with a rocker switch and the wheels snap to the stepper home position. Marv is then placed in the home circle and aligned "square" by eyeball. When the grain of wheat light goes out, Marv is started with the buzzer.

It moves to the end of the hall, turns left and goes to the room B doorway. The right UV detector measures for 2 seconds. If the count is high, the candle is in the room; if it is low it isn't. If the candle is not in the room B, Marv backs up to the room A doorway and measures with the left UV detector for 3 seconds. If the candle is not in A, Marv backs up to end of the transverse hall and searches room C with the left UV detector for 2 seconds. Not there? Proceed to the doorway of room D and search with the right UV detector. Still not there? Back to room C, this time moving directly to the doorway and measuring with the left UV detector.

If/when the candle is found, Marv pivots to face the doorway



Sound activation board with microphone protruding from left.

and scans with the PIR detector. If not found, Marv advances inside the doorway and scans again. If not found, Marv advances again while monitoring the bumper switches. If the bumper *is not* struck, Marv scans once more from the new position. At whatever scan spot the

Above: Silver Bullet *discovers Kate Murphy at AAAI-97.*
Left: Bujold *articulates to three positions.*

win-stay" foraging behavior of squirrels and bees. Additional code for frontier-based exploration algorithms was donated by the Naval Research Center's AI Laboratory. This programming enables *Silver Bullet* to navigate autonomously outdoors over uneven terrain at speeds up to 8 mph, carrying a 60-pound payload (*Bujold*).

Silver Bullet's size and car-like steering prevent her from going into tight corners or climbing over rubble. Nevertheless, she can quickly drive around the perimeter of a disaster zone looking for likely places to investigate. She can do this autonomously, using her onboard Pentium computer and AI software to process information from her camera, microphones, sonars, thermal probe and inclinometers to navigate safely and search out possible victims.

The Sensors

Silver Bullet has a healthy variety of sensors. A camcorder is mounted on a panning mast (controlled by a CANAMP module). An E2T thermal probe is also mounted on the panning mast. The thermal probe, designed for medical applications, can detect changes in temperature up to 20 feet away. This allows the robot to scan an area for warm spots relative to the ambient temperature. If a "warm body" is located, *Silver Bullet* alerts the teleoperator, who can then use the video feed of the camcorder to determine if the warm spot is, indeed, a survivor. *Silver Bullet* also carries a microphone and speakers so the teleoperator can communicate with the victim.

Silver Bullet has six sonars and a homemade inclinometer for navigation. The sonars are Polaroid Lab grade II ultrasonics controlled by a 68HC11 microprocessor control unit (MCU) programmed in assembly

continued on page 52

The Vx-Files:

What the Media Couldn't Tell You About Mars Pathfinder

byTom Durkin

Arguably the most spectacular interplanetary robot mission in history, *Mars Pathfinder* outperformed all expectations – and ironically, that was why the lander developed a mysterious communications problem shortly after its successful landing July 4, 1997.

For no apparent reason, *Pathfinder's* onboard computer would spontaneously reset itself. This happened about a half dozen times in the first few weeks after the landing.

Contrary to press reports at the time, no data was lost, but data collection was delayed by a day every time *Pathfinder* reset itself.

"Reports of the problem in the popular press were generally incoherent, and in some cases wildly wrong," according to David Wilner, chief technical officer of

Photo courtesy of NASA.

Flight software engineer Glenn Reeves had reason to grin after he and his team solved Pathfinder's *"software glitch." Behind him is the duplicate Mars lander that the JPL/Wind River team used to find and fix the priority inversion problem in the VxWorks® program. "The biggest thanks should go to the software team that I had the privilege of leading. Their expertise allowed us to succeed," he emphasized.*

Wind River Systems of Alameda, Calif.

In a Dec. 3, 1997, keynote address to the 18th IEEE Real-Time Systems Symposium in San Francisco, Wilner explained that the media couldn't grasp the complexity of the problem – much less the answer – "because of the difficulty of explaining the issues to non-engineers."

However, because he was speaking to real engineers Dec. 3, Wilner detailed precisely what went wrong last summer, why it went wrong – and how it got fixed. Wilner was uniquely qualified to speak on the subject, because Wind River makes the RTOS (real-time operating system) – VxWorks – which was embedded in *Pathfinder's* onboard computer.

Fortunately, Wind River and NASA software engineers were able to replicate the problem on a duplicate of *Pathfinder* at the Jet Propulsion Laboratory in Pasadena, Calif. It only took 18 hours to fix the problem – but it took almost three weeks to find it.

People Will Talk

Michael Jones, a researcher in the Operating Systems Research Group at Microsoft Corp., found Wilner's speech so interesting that he wrote up a report on it and e-mailed it to some friends, colleagues and educators.

Within days, his report was posted on several Usenet interest groups, where it was widely redistributed throughout the larger Internet community.

"It was actually a pretty good take on my description," Wilner told *RS&T*, adding, "One of the guys at JPL posted a reply to Jones that went into a lot more gory detail about what happened from their point of view."

The "guy" at JPL was Glenn Reeves – the flight software cognizant engineer for *Mars Pathfinder*. Reeves led the team that identified *Pathfinder's* reset problem and fixed it.

Using Jones' report as a framework for his response, Reeves elaborated – in painstaking detail – the exact software engineering methodology his team went through to replicate, and then rectify, what the general media called a "software glitch."

How VxWorks Works

In order to understand what went wrong, it is necessary to understand what VxWorks does. Basically, the software provides "preemptive priority scheduling" of data threads and mission control commands.

Data (video images, soil samples, meteorological readings, etc.) from the various instruments on the lander (*Pathfinder*) and the rover (*Sojourner*) had to pass through an information bus in *Pathfinder's* computer to be transmitted to Earth.

Likewise, commands to control the devices on *Pathfinder* and *Sojourner* (such as the cameras or alpha proton X-ray spectrometer) had to move through the same information bus in the opposite direction.

Obviously, this couldn't happen all at once. Data threads and command strings had to take turns using the bus. Furthermore, some data and commands were more important than others. Thus, it was the job of VxWorks to schedule traffic through the bus according to the pre-assigned priorities of data and commands.

In order to prevent communications conflicts, VxWorks synchronized access to the bus with mutual exclusion software locks known as "mutexes" or "semaphores." When a specific task was running, its semaphore blocked other tasks from interfering with it.

"A Classic Case of Priority Inversion"

"Most of the time this combination worked fine," Jones wrote in his report. However, very infrequently, an interrupt was sent to the bus that caused a medium-priority communications task to be scheduled during the split-second interval when a high-priority thread was blocked while waiting for a low-priority meteorological data thread to run.

In this case, Jones explained, the long-running, medium-priority communications task – having a higher priority than the low-priority meteorological task – would prevent the meteorological task from running. After a predetermined time had passed, a watchdog timer would go off, notice that the low-priority data bus task had not been executed on time, conclude that something had gone wrong, and initiate a total system reset.

"This scenario is a classic case of priority inversion," Jones asserted.

"Priority inversion is a difficult concept to explain," Reeves told *RS&T*. "It doesn't make any sense in everyday terms. In a nutshell, what happened is there was a situation where the act of taking the information took longer than the amount of time we had allotted for the rest of the communications to occur within the software time limits."

Too Much of a Good Thing

What caused the priority inversion was that *Pathfinder's* antenna performed better than expected.

"It turned out that we got a much higher meteorological data rate, because we could point the antenna at Earth much better than we ever imagined," Reeves said. "We didn't expect it. We had never actually tested the thing with that high a set of data rates. It was better than the best possible hope we had.

"It was really great news from the spacecraft, but it was somewhat disconcerting to us, because it did expose a bug that we really wish we had caught."

Bug Busters

"The software that flies on *Mars Pathfinder* has several debug features within it that are used in the lab but are not used on the flight spacecraft (not used because some of them produce more information than we can send back to Earth)," Reeves wrote in his response to Jones.

While not enabled on the spacecraft, these features remained in the software by design. "We strongly believe in the test-what-you-fly-and-fly-what-you-test philosophy."

One of the debugging tools was a trace/log facility that was originally developed to find a bug in an early version of VxWorks.

"After the problem occurred on Mars, we ran the same set of activities over and over again in the lab," Reeves said. The trace/log was coded to dump its record of activities when a failure occurred. Finally, "we were able to cause the problem to occur. Once we were able to reproduce the failure, the priority inversion problem was obvious."

What was not so obvious was the solution ...

The Big Fix

As both Jones and Reeves explained it, the "priority inheritance" parameter of the semaphore for the meteorological data thread was not enabled in *Pathfinder's* VxWorks software.

If priority inheritance had been enabled, "the low-priority meteorological thread would have inherited the priority of the high-priority data bus thread blocked on it while it held the mutex, causing it be scheduled with higher priority than the medium-priority communications task, thus preventing the priority inversion." Jones wrote.

Clearly, the JPL flight software team had to "change the creation flags for the semaphore so as to enable the priority inheritance," Reeves explained.

continued on page 53

Basics of a Digital Brain: Part 1

by veteran robot programmer ***Karl Lunt***

You want to build a robot, but not just a simple robot. In just a couple of hours of blue-sky contemplation, you've decided to combine several motors, some sensors, and tons of intelligent behaviors. But now, you've hit the big challenge: How do you control all of these elements so that

YOUR ROBOT ACTUALLY DOES SOMETHING!

No doubt about it, you need a "microcontroller."

If you've never worked with a microcontroller before, then this article is for you. I'll start with the "black-box" view of a microcontroller, and when I'm done, you'll begin to see that *microcontrollers are tools*.

When you view a microcontroller as a building block, like a motor or a battery, you'll be on your way to designing more complex and robust machines.

The Black Box View

First, we'll limit ourselves to seeing what goes in and what comes out, without looking into the microcontroller itself. This is called the black box view.

Viewed simply, a microcontroller (MCU) is an electronic device that merges a small computer, some memory, and a collection of input/output (I/O) subsystems, all on a single chip. Once you've installed a program into the MCU's memory and the program runs, the MCU will sense voltage conditions on its input lines, make decisions according to its program, and alter the voltage states of its output lines.

Introducing Hexes, Bytes, and Addresses

Let's first consider the MCU's computer. (I'll assume that you know that a bit is a small voltage that is represented by a 1 or a 0, and that a chunk of 8 bits are called a byte.) All MCUs have at their core a small computer, usually an 8-bit device that can process a byte at a time. Since each bit must be either 0 or 1, the eight bits within a single byte can form 256 different patterns. Here are equivalents in the binary, decimal and hexadecimal numbering systems:

```
00000000 = 0 decimal or $00 hexadecimal
00000001 = 1 decimal or $01 hexadecimal
00000010 = 2 decimal or $02 hexadecimal
00000011 = 3 decimal or $03 hexadecimal
...
11111110 = 254 decimal or $fe hexadecimal
11111111 = 255 decimal or $ff hexadecimal
```

The above table shows two crucial points to remember:

First, programmers count differently than the rest of us: They count beginning with 0, not 1. For instance, if I were to refer to the third element in a span of memory, I'd be referring to memory cell #2, not #3.

Second, most of the time I'll be using the hexadecimal numbering system. This system is simply more convenient to use when talking about 8-bit numbers. The hexadecimal math system uses the digits 0-9 and a-f. I won't go into more detail than that here, but I'll try to make my later examples easy to follow so those of you who haven't seen hex before can keep up.

As I said, an 8-bit MCU handles information in bytes. Generally, these bytes of data occupy fixed locations in the MCU's memory. Such locations are known as memory "addresses." How many bytes an MCU can access at once depends on the number of address lines it has available. Many common MCUs, such as Motorola's popular 68HC11 family, sport 16 address lines. This means they can access memory addresses that range from $0000 to $ffff. Refer to my discussion of possible eight-bit patterns above, then expand the table to use 16 bits and you'll see what I mean.

If you convert these numbers to decimal instead of hexadecimal, you get addresses that range from 0 to 65,535. The latter number is usually shortened to 64K, which really means 64 times 1,024 bytes; (This illustrates that a kilobyte is really *not* just 1000 bytes, but is actually *1024* bytes. This convention is used because 1024 is an even power of 2, but 1000 is not.)

Note that *being able to access* 64K of memory does *not* mean that an MCU *actually has* 64K of memory on the chip. Microcontrollers come with a small amount of memory already on the device. But if your program needs more memory than comes on the chip, you have to add the extra yourself, which entails adding electronics.

Remember that MCUs also include on-chip I/O elements. The 68HC11 chips reserve a small set of addresses in memory for controlling the I/O elements. To your program, these I/O addresses, known as I/O registers, behave like memory. But your program can write a value to one of these I/O registers and cause electronic signals to change on various MCU output pins. Similarly, changes that occur on some of the MCU pins can be detected by your program when it reads one of these I/O registers.

Thus, the I/O registers represent the interface between your program and the outside world.

I/O registers can also be used to access the complex subsystems *within* the MCU. For example, the 68HC11 MCUs all contain a sophisticated set of timers and counters, which your program can use to time events or to generate accurate pulses. Your program can control how these timers behave, by accessing a dedicated set of I/O registers tied to the timer subsystem. Other 68HC11 I/O registers control features such as serial ports, analog-to-digital (A/D) converters, and even lower level characteristics of the MCU itself.

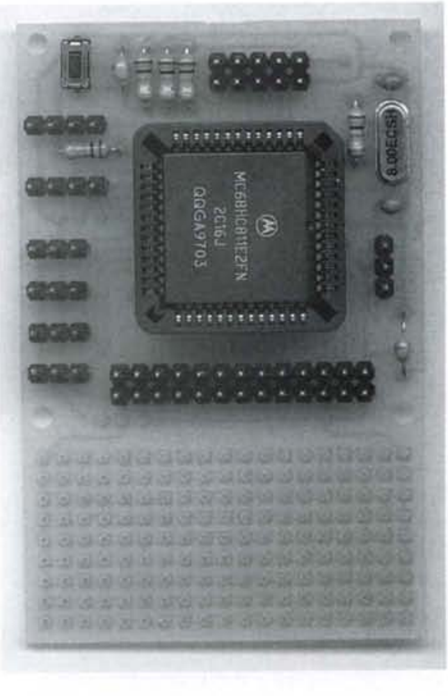
But viewing an MCU as a black box with I/O registers and memory leaves out a vital element: execution speed. The MCU is always running some form of a program, and is running it very quickly. The 68HC11, while slow by today's standards, still performs two million memory accesses per second. This means the machine is doing something at the rate of about one million instructions per second. Your job, as the robot's programmer, is to make sure that the MCU is executing *your* program while it is running. (I guarantee it will be executing *something*: If it's not running your program, then it might be executing the senseless bytes in random contents of memory, which is *not* a Good Thing.)

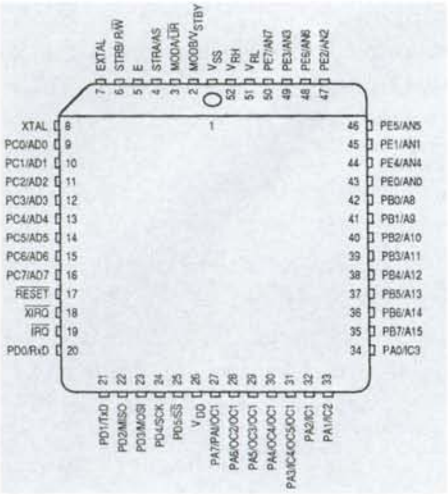
Power Up! Pointers and Vectors Explained.

So how does an MCU actually get started, and how does it decide what to? I'll continue using the 68HC11 as an example: When you throw the power switch to start the 68HC11 running, the MCU's computer examines the byte stored at address $fffe and the byte stored at $ffff. It merges these two bytes together to form a single, 16-bit "word." It then interprets this word as a new address and then reads the byte it finds at that address. The MCU uses *this* byte as the first instruction it executes. The results of that instruction guide the MCU to the next instruction, and the sequence continues as the MCU runs its program.

This technique of using the contents of one address to define another address is known to C

Left: BotBoard 2
Below: BotBoard 1
These examples are populated with Motorola 68HC11 MCUs.





Pinout of the MC68HC11E2 52-pin PLCC. Also comes as 48-pin DIP.

programmers as a "pointer." Assembly language programmers call it a "vector." In fact, 68HC11 programmers refer to the addresses $fffe and $ffff as "the reset vector," since they form the 16-bit vector that the MCU uses immediately following reset or power-up.

This concept of a reset vector lets you place your program anywhere in the 68HC11's memory space, yet tell the MCU exactly where it can find your program after power is reset.

Overview of an MCU

First, you need to know that RAM memory is *volatile*: It "forgets" its contents when power is turned off. But EEPROM memory is *non-volatile*. EEPROM stands for Electrically Erasable Programmable Read-Only Memory. This is simply a lengthy term meaning, "memory you can store programs in, even after the power is turned off."

The 68HC811E2 can store up to 2,048 bytes of program code in EEPROM. It can also *temporarily* store 256 bytes in RAM as long as the power is still on.

Besides 2K of EEPROM and 256 bytes of RAM, the '811E2 also contains a good assortment of I/O subsystems:

• An asynchronous serial port, called the SCI, that you can use to talk to serial devices such as your PC's COM port.

• A synchronous serial port as well, dubbed the SPI. This high-speed link can move data between the MCU and other chips at up to one million bits per second.

• Eight channels of 8-bit A/D (analog-to-digital converters), which allow your program to sample analog voltages on any of eight pins and translate those voltages into digital information.

• Eight timer channels and large a number of digital I/O lines.

The 68HC811E2 has nearly everything a robot builder would want.

A 68HC11 chip looks like a thick, black plastic postage stamp. Most come in a plastic leaded chip carrier (PLCC) package with 52 pins. To make these devices actually do something useful, you must first connect them to your robot circuitry.

Getting Started With a BotBoard

To get started using a 68HC11 right away, nothing beats a BotBoard. Marvin Green, a friend of mine in Portland, Oregon, designed the BotBoard some time ago as a simple way to get a single-chip 68HC11 system up and running. This little printed circuit board (PCB) is smaller than a playing card, yet has all the paths laid out for the basic 68HC11 system. You can buy a blank BotBoard from Mondo-tronics (www.robotstore.com) for about $10. You will also need to find and order the remaining parts, including the 68HC811E2 MCU device and a matching 52-pin PLCC socket. The BotBoard documentation includes a full parts list with suggested suppliers. Expect to spend about $50 in all to build up your first BotBoard. (For suggested sources of components and software, see the Resources section at the end of this article.)

The BotBoard instruction book also includes plans for a simple serial cable that you will need to build. This cable is needed because the BotBoard's serial port talks with digital-level signals (0 to 5 volts), but your PC COM port talks using RS-232 levels (-15 to +15 volts). If you connect the two devices together without using the suggested converter, you will damage or destroy your BotBoard. The cable and converter are easy to build, and you will be able to use them on many different BotBoard projects.

For a power supply, simply grab a four-cell AA battery holder from Radio Shack and fill it with alkaline batteries. I really like the Renewal rechargeable alkalines, and I use them for all of my robot projects. Whenever the total voltage from my four batteries drops below five volts, I pull the batteries out and stick them in the charger. Note that you *must* use a Renewal charger to recharge Renewal batteries: using any other kind of charger is asking for a possible fire or explosion. Also, recharge Renewals as often as possible. Unlike standard NiCads, Renewals don't form a "memory," and they last longer when recharged frequently.

The BotBoard makes an excellent robotics tool, but it is complex, so naturally you need tools to use the tool. For the 68HC11 in general, those tools include software like assemblers, compilers, and communications programs. Mondo-tronics sells a couple of floppy discs that will include most of the tools that you will need for working with the 68HC11.

You can easily buy the software you need. But for those of you who are interested in gathering up your *own* tool set, I'll now briefly describe what you'll need and where you can get it. Reliable sources for this software are also detailed at the end of this article.

First off, you will need a copy of a program called "pcbug11." This Motorola software runs on your PC and talks to the 68HC11 from the PC's COM port into the 'HC11's SCI.

REALIZE THAT NONE OF THIS IS OVER YOUR HEAD; JUST SPEND SOME TIME WORKING WITH THE CIRCUIT AND LOGIC PROBE, AND RE-READING THIS ARTICLE AS NEEDED

Though written in the days of DOS and somewhat temperamental on today's faster machines, pcbug11 is still both indispensable and great fun. You can use this program to probe areas of the 68HC11's memory, write and debug small programs, modify the MCU's I/O registers, and even do limited tracing and debugging of your programs.

To find a copy of pcbug11 on the Internet, aim your web browser at Motorola (www.mcu.motsps.com:80/freeweb/) and follow the links that take you to the 68HC11 area. Download the program pcbug342.exe, copy it to a new directory called pcbug11 (or suchlike), and execute the program. It will expand into a working version of pcbug11 with all of its necessary support files.

Second, you will need programs called an "assembler" and a "compiler" to help you create code for the 68HC11. The other articles in this series will use my homebrew SBasic compiler for examples, so I suggest you stop by my web site at www.seanet.com/~karllunt and hit the Tips and Techniques page. There you will find a copy of my SBasic distribution file; this file contains the SBasic compiler and a serviceable assembler, plus a lot of sample programs. You will also find a link to a page explaining in detail how to use pcbug11 to talk to a BotBoard. Download that page and file it away for reference.

Third, you absolutely must have a copy of Motorola's *M68HC11 Reference Manual (M68HC11RM/AD)*, known in the microcontroller world as the "pink book" because of the color of its cover. You can usually order this book from the nearest Motorola distributor or sales representative; you can also contact Motorola directly by using the phone number for their literature center, available on their web site.

Note that this book is vital for successfully designing with the 68HC11; if you don't have this book, you doom yourself to endless frustration.

Last, but not least, you will need tons of information. The Motorola web site listed above is a great starting point. Another good place to begin is my web site, as well as sites for which I've provided links. You can also do a web search and locate a copy of the 68HC11 Frequently Asked Questions list, known as the 68HC11 FAQ. Some of the info is a little dated, but it still contains a lot of meat and will answer many of your first questions.

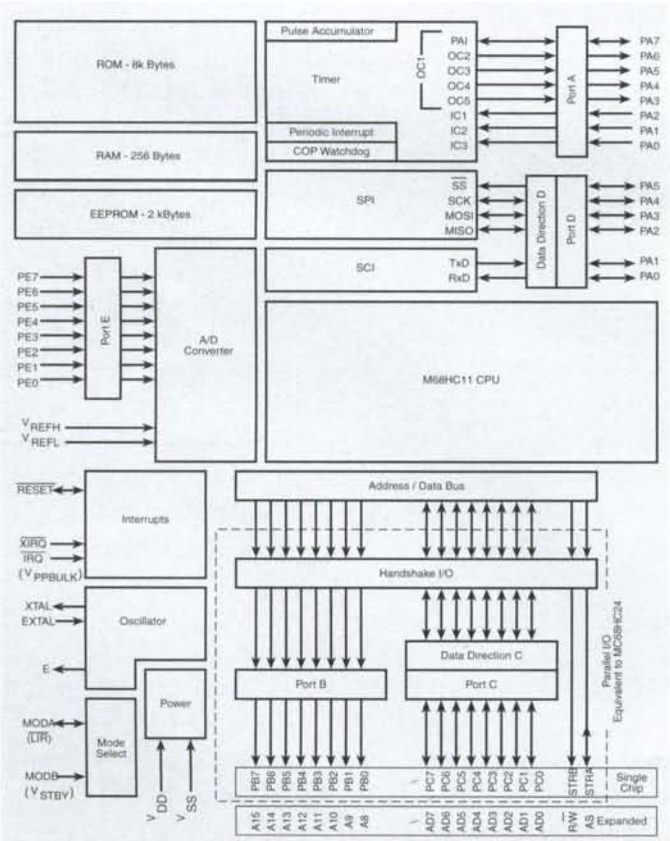
Elements of an MCU

The 68HC11 I/O subsystems let your program interact with the outside world, and it's vital that you understand how they work so you can use them effectively. Refer to the accompanying functional view of the 68HC811E2.

This MCU, like most of the 68HC11 variants, contains five major I/O ports, each connected to the outside world by pins on the device's package. The ports are known by the alphabetic letters A through E, and by their major functions.

Port A consists of pins PA0 through PA7, connected to the 68HC11's 16-bit timer subsystem. Generally, the 68HC11 can use PA0 through PA2 as counter inputs; that is, the MCU can sense changes on these three pins and make that information available to your program. Additionally, pins PA3 through PA7 can output timing signals generated by your program to control motors or other actuators.

Functional Block Diagram: 68HC11E2



If you don't need to use any of the timer-based functions on the port A pins, your program can still use PA0 through PA2 as digital input lines and PA3 through PA7 as digital output lines.

Port B consists of pins PB0 through PB7, connected to the 68HC11's multiplexed address and data buses. (A bus is simply a group of conductors that are used together to pass signals.) These buses let you add external RAM or EPROM to your system, to gain more memory than is available in the 68HC11 chip. Such memory expansion is beyond the scope of this article, but is discussed in the books mentioned above and is well worth study.

Since you won't be using the external addressing on your BotBoard designs, your program can use PB0 through PB7 as digital *output* lines. Note that they *cannot* be used as digital *input* lines.

Port C consists of pins PC0 through PC7, also connected to the 68HC11's multiplexed address and data buses. Again, since you won't be using external memory on your BotBoard designs, your program can use PC0 through PC7 as digital output lines.

Unlike port B, however, any or all of the port C lines can also serve as digital *inputs*.

Port D consists of pins PD0 through PD5, connected to the SCI and the SPI serial subsystems.

The 68HC11 uses PD0 and PD1 to receive and transmit serial data asynchronously, in a format compatible with the PC's COM port. These lines make up the SCI pins.

PD2 through PD5 form the *SPI* pins and are used for high-speed serial communications with special chips having a compatible interface. Your designs will almost always use PD0 and PD1 for RS-232 communications.

If you don't need the SPI functions, your program can use PD2 through PD5 as either digital inputs or digital outputs.

Finally, **Port E** consists of PE0 through PE7. These pins connect to the 68HC11's A/D subsystem. You can connect a voltage from 0 to 5 VDC to any of these eight pins, and your program can then read the voltage on that pin as a value from 0 to $ff. Each of the eight port E pins works independently of the others, so any pin that you don't need as an A/D input can also act as a digital input.

As you can see, the 68HC11 provides you with 38 I/O lines that you can use in a variety of ways to control your robot. Lines PB0 through PB7 are the simplest form of 68HC11 I/O, so I'll use them as my first example. Refer to the accompanying diagram on writing data to port B.

This example shows what happens if your program writes the value $32 to address $1004. In the 68HC811E2, address $1004 isn't a memory location, though it behaves like one as far as your program is concerned. Actually, the eight bits in address $1004 make up the eight digital output lines of port B. The value written to bit 0 of $1004 appears directly on the pin PB0; the value of bit 1 appears on PB1, and so on. Writing a 0 to one of these bits causes the MCU to set the corresponding digital output line to a logic 0, or ground. Writing a 1 to a bit causes the MCU to bring the digital output line to a logic 1, or +5 VDC.

Port B (or PORTB, to use the Motorola convention) makes a great starting point for investigating how a 68HC11 works: The port requires no setup of any kind; just write a value to it and that value appears on output lines PB0 through PB7. The MCU isn't even fussy about how you write the value, just as long as something gets written there. So you can use pcbug11 to alter the value in address $1004 and see the effects of your change immediately, using a logic probe or simple LED circuit. To use pcbug11 to change port B, use the command sequence:

```
control base hex
ms 1004 32
```

Note that you only have to enter the control statement once, when you first start up pcbug11. It remains in effect from that point on.

WARNING: The PORTB outputs, like the other 68HC11 output port lines, cannot provide a lot of power, so you need to use caution when you hook up external circuitry. For example, a PORTB line gets set to +5 VDC when you write a 1 to the corresponding bit, but the 68HC11 can only supply about 1 mA of current at that voltage. If your circuitry tries to draw more current than that from the output line, the voltage will drop below +5 VDC. If your circuitry tries to draw more than about 20 mA of current from any output line, you risk damaging the 68HC11.

The bottom line here is use care when designing your external circuitry. You cannot drive devices such as relays directly from 68HC11 output lines, so don't try. LEDs with suitable dropping resistors (say 240 ohms or more) are fine, but devices that need higher currents call for some kind of interface IC in between. I'll cover some suitable interface chips in future articles.

Port C (PORTC, address $1003) works almost exactly like PORTB, but you can declare some of the port C lines as digital inputs and some as digital outputs. This declaration takes place using an I/O register named DDRC (address $1007). Before your program tries to use PORTC, it should first declare the directions of all PORTC lines, so you don't inadvertently change some external circuit. The value your program writes to DDRC determines the direction of each PORTC digital line. Writing a 0 to a bit in DDRC makes the matching bit in PORTC an input line; writing a 1 to a bit in DDRC makes that line an output instead.

Once you've set up DDRC, your program can change the state of the PORTC output lines by writing a value to PORTC, just as I've described for PORTB above. Any PORTC lines that have been configured as digital inputs will ignore the change; those lines remain inputs to the outside world. Similarly, your program can read PORTC to determine the current state of those pins configured as inputs. When your program reads a PORTC input line, the corresponding bit will be 1 if that line is connected to +5 VDC, or 0 if that line is connected to ground. Reading any output pins will return the value most recently written to them.

My last recommendation is this: Realize that none of this is over your head; just spend some time working with the circuit and logic probe, and re-reading this article as needed.

Build your BotBoard using the supplied instructions, load up pcbug11, and spend some time playing with the 68HC811E2 chip. You can buy a logic probe just about anywhere, and they make great tools for finding simple hardware problems and for checking the state of your 68HC11 output lines.

Next time, I'll cover some more of the 68HC11 I/O subsystems, and get you started with your first program. I suggest that you print out the SBasic user's manual that comes with the SBasic distribution file and get started by reading that. I'll use SBasic for my programming examples, and this head start will pay dividends. -Karl RS&T

Someday soon, you'll want to build a sturdy metal robot. In this fact-packed briefing, veteran Robot Warrior Ronni Katz helps you understand aluminum: selecting grades, cutting, bending, welding and bolting.

Fabrication: The Basics

Winning With Aluminum

Story and photos by ***Ronni Katz***

Understanding the Metal Itself

Pure aluminum is not heat-treatable, but it gains strength when bent, hammered or otherwise worked. However, aluminum is also subject to fatigue. This is why, even though it has excellent conductivity, it is a poor choice for wiring. Repeated bending of aluminum wire will cause it to weaken, crack and lose its ability to carry current. However, when used in the fabrication of parts for robots and other structures, aluminum has many advantages. For one, it is as workable as steel but does the job for a fraction of the weight and cost.

Some of aluminum's alloy groups provide strength similar to that of mild steel.

Alloying aluminum does not adversely affect its work-hardening characteristic, but the addition of elements to aluminum affects the metal in other respects. One major alteration is that some alloys are heat-treatable, which is important to consider when welding is to be a part of the metalworking project. Also, some of the alloys offer corrosion resistance, an important factor if the project is something that will be exposed to the elements.

Aluminum and aluminum alloys come in a variety of shapes and sizes – tubes, bars, I-beams, sheets, etc.

Choosing the Right Alloy

Aluminum comes in nine different series of alloy groups. The 1000 series is almost pure aluminum, with minute percentages of impurities indicated by the lot number. For example, Alloy 1345 = 99.45 percent aluminum. The "1" is the series number, and the "3" indicates modifications in impurity limits. If the second number were a zero, then there is no special control on individual impurities. The last two digits indicate the minimum aluminum percentage expressed to the nearest 0.01 percent. The 2000 to 8000 series are alloys of aluminum where the base metal is joined with other elements. Mostly what I will be dealing with here are the 5000, 6000 and 7000 series, which are the ones most often used for metalworking projects by hobbyists whose projects require good strength, workability and resistance to corrosion.

The 5000 series is what is most frequently sold in Home Depot and similar do-it-yourself stores. This is architectural-grade aluminum. It provides strength, fine welding qualities and good corrosion resistance. The most common alloy is 5052, which is used a lot in home construction for window and doorframes. The Navy has its own alloy – 5086 – that is used in shipbuilding, because it is strong and highly corrosion resistant.

Two quick ways to identify the 5000 series from 6000 and 7000: If the piece is an I-bar or like shape, on the 5000 series it will have an inside corner that has a sharp, precise edge. In the 6000 and up series, the inside of the corner will have a rounded edge. This is because the 6000 and above series are extruded, and this process causes the bar to have that rounded inner corner.

The second quick trick is that aluminum sheets in 6000 and higher will usually have the series stamped on the metal. For example, I had a piece of aluminum with both the Mil Spec Q number on it as well as the aluminum factory 6061 T4 notation on it.

Although not as hard as the 6000 and 7000 series, the 5000 series is easier to obtain and less costly than the higher aluminum grades. If the 5000 series isn't strong enough for you, the 6000 series is heat-treatable, possesses good formability and corrosion resistance, and has good strength. The major alloy is 6061, and it is one of the most versatile heat-treatable alloys.

The 7000 series has zinc as the major alloying element. When coupled with a little magnesium, the result is a heat-treatable alloy of very high strength. The 7075 alloy is the best member of this group. It is used for the construction of airframes and other highly stressed parts. It is obtainable by the hobbyist, but it is more costly because it is aircraft-grade aluminum.

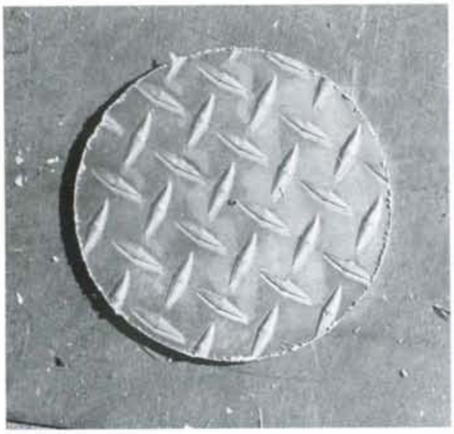
Selecting a Shop to Help You

The 6000 and 7000 series are not readily procurable through building supply stores. Fortunately, places like Small Parts Inc., a surplus parts store based in Miami Lakes, Fla., might have odd lengths of the material. If not, metal-supply places will sell aluminum to the hobbyist.

The problem with such metal wholesalers, however, is that they require a *minimum* order. This may force you to buy far more than you need. For this reason, I recommend finding a metalworking or machine shop in your neighborhood. They might have small lots of the grade of aluminum alloy you're looking for, and they might be willing to sell some to you for a better price than you might get from a metal supplier.

Central Metal Fabricators, Inc. – the shop that helped me in the construction of my robot *Spike II* – has an impressive selection of various aluminum alloys that they purposely keep on hand for local small-users, like hobbyists, sculptors and such. If your project requires superior strength, then I recommend you seek out your local metal shop, or Small Parts, to try to obtain the aluminum alloy you need.

> SOME OF ALUMINUM'S ALLOY GROUPS PROVIDE STRENGTH SIMILAR TO THAT OF MILD STEEL.



A plasma cutter allows the designer to create a wide variety of shapes, including the circle shown here. The edges will be rough and need finishing. Nevertheless, a plasma cutter is the tool to use for cutting unusual shapes.

Understanding Aluminum Work

Aluminum has some advantages over steel. One advantage is that it is available in a variety of shapes and thicknesses. It comes in bars, tubes, sheets, etc.

A second advantage is that it is easier to machine than steel. Steel requires fabrication; aluminum does not. A shape can be designed on a computer using CAD (computer-assisted design) software and then be programmed into an automated milling machine.

At Central Metal Fabricators, I saw how they took a solid plate of aluminum and milled it to exact specifications. The finished work was *precisely* the size and shape the customer wanted, and it was made out of a solid piece of aluminum – no welding. Thus, with aluminum, it is possible to go from CAD to final cut with a lot less labor cost than steel cutting.

An important point to note when cutting aluminum: Don't use abrasive cutting wheels. It will be like trying to cut through wax. Aluminum is softer than steel and will "plate" onto an abrasive wheel. At best, it will clog the cutting teeth and make it more difficult to cut the metal. At worst, it will get into the bridges between the teeth, expand from the heat of friction, and pop the teeth off. Use WD-40 to lubricate the cutting wheel and keep the aluminum from sticking.

Band saws and plasma cutters (a cutting torch) are best for getting precise cuts, although commercial mills usually use big radial-arm saws to cut aluminum. Using a band saw, it is possible to get a nice clean edge that will need minimal finishing. Amateur builders may not have access to a plasma cutter, but these devices give you the ability to get interesting shapes, such as circles, out of aluminum.

SOME SHOPS HAVE A SELECTION OF VARIOUS ALUMINUM ALLOYS THAT THEY PURPOSELY KEEP ON HAND FOR LOCAL SMALL USERS, LIKE HOBBYISTS, SCULPTORS AND SUCH

Regardless of the type of saw, one thing to note is the spacing of the teeth. Bigger spaces are better – they are less likely to clog the blade.

For grinding aluminum, it is best to use an aluminum abrasive wheel, which is less likely to clog, and a cutoff wheel. The down side of using these two wheels is that they wear out more quickly. However, the fact that they'll wear out before they clog up is actually a positive aspect.

Here's a trick for maximum efficiency and ease for sanding your aluminum project: Wax the sanding disk.

Most Home Depot stores sell a paraffin-based wax you can use to coat your sanding disk. We used Edge Lube, a commercially available wax lubricant, on the sander to work on *Spike II*. Wax enhances the sanding process by keeping the aluminum from sticking to the disk (which makes the grinding less of a grind).

A lathe can be used to "face off" a piece, giving it a nice finished look.

But a word of caution: If during the lathing process you hear what has been nicknamed "lathe music" – a high-pitched note or whine – stop the lathe. Lathe music is caused by the aluminum vibrating on the tool. If not prevented, the vibrations will cause noise marks – a clear sign your work has been damaged.

Can You Do It Yourself?

For small projects, a Dremel tool (www.dremel.com) is the tool of choice. It can cut, grind, mill, polish, burnish and sand your aluminum. Much of the detail work on *Spike II* was done with a Dremel. But the major cutting, welding and bending should be done by professionals.

Bending

If your design calls for the metal to be bent, you need to know that

WITH ALUMINUM, IT IS POSSIBLE TO GO FROM **CAD** TO FINAL CUT WITH A LOT LESS LABOR COST THAN STEEL the *temper* of the aluminum will affect its bendability. A T6 piece is the hardest temper you can get, but a high temper piece doesn't necessarily mean it will be unbendable. It just means it will be harder to bend. Most pieces I used were T4 and quite bendable.

Also, you should know that, unlike steel, heating aluminum *doesn't* make it more pliable. Heating aluminum will make it break quicker. If you are going to bend aluminum, cold-work the metal.

Cold-working means doing the work at standard room temperature, not outside in North Dakota in January. Very cold temperatures also make aluminum more likely to break under the stress of the bending process.

A phenomenon to note when you're bending aluminum is that as you bend it, you will feel it getting harder.

Caution, it gets hardest just before it breaks. Furthermore, if it's going to break in the direction of work, then it's going to break when you bend it back to its starting point as well. The idea here is to get your piece bent to the angle you need without having it go brittle on you and snap.

It's a good practice to take a test piece and bend it to the desired angle. If the bend looks solid and there are no signs of metal fatigue, then you can be pretty sure it will be safe to do the bend on the project piece.

Two related aspects to take into consideration when working with aluminum are surface tension and stress relief. The rolling process of manufacturing sheets of aluminum and the like squeezes the surface tighter than the edges. There is surface tension on all of the material but it is mainly around the skin of the metal, not so much toward the center of the piece. "Stress-relieved" aluminum is absolutely necessary when doing an application where precision is required. Using this type of metal will insure that after you cut, weld and assemble, your aluminum will meet your design specifications.

This one-inch bar of 6061 aluminum was used to make test bends. The test was successful, meaning it was safe to use this alloy on the actual project under construction.

Here's an example of what can happen if your metal is not stress-relieved: Let's say your design calls for a series of wheel pockets on one side of a bar, while the opposite side of the bar is to be solid. If the bar is not stress-relieved, the bar will bend slightly, because you have relieved all the surface tension off the side with the pockets, and the solid side has pulled tight. How much the bar will bend depends upon the particular piece of aluminum you're using.

To prevent such bending, you must use stress-relieved aluminum. Stress-relieving aluminum isn't cheap, but if your application calls for absolute precision, you can't afford not to use it.

Welding

Welding aluminum is *not* like welding steel. For starters, aluminum requires a high-frequency, alternating current (AC) to break down the oxides that form on the surface as the metal is being heated. The oxide that forms on the surface of the heated aluminum is nonconductive and has a higher melting temperature than the aluminum itself. If the oxide layer is thick enough, the aluminum under the oxide will start to melt while the oxide remains solid above it. Therefore, you need high-frequency AC to break down this oxide.

The need for high-frequency AC makes welding aluminum at home difficult for a hobbyist. Another down side is that you must control the voltage while welding aluminum. With steel, you just set the machine, get the arc going, and you're ready to rock. Not so with aluminum. You need a lot of amperage to get the material hot, because of the metal's high heat conductivity. (It is this property of aluminum that makes it the metal of choice for the construction of heat sinks.)

In order to weld your piece, you have to flood it with heat until the area you are working on gets to the point that it cannot dissipate the heat fast enough. Once you get the metal to where it needs to be, you must back off the current. You can't just "set and forget," the way you can with steel. Because you have to go up and down with the current while working, welding aluminum is tricky.

Another challenge in welding aluminum is the shielding gas. Some welders require straight argon. There is, however, another option: the heliarc torch. This torch uses helium as its shielding gas, because helium increases the rate of heat transfer. A spool-fed torch loaded with 4043 aluminum wire works well for most welding projects. You can also use 5356 wire, but it produces much more smoke, because of the higher levels of magnesium in the aluminum alloy wire. If you're going to use 5356 as your filler metal, make sure you have adequate ventilation.

A mig welder is great for large projects that require thick welds and for times when a quick weld is needed. Although with a mig welder it is more difficult to do the really fine stuff, it isn't impossible to get a good clean weld if you know what you are doing.

If you don't have access to a coil-fed or spool-fed welder, you can use the tig welding technique – wire in one hand, torch in the other. When tig-welding, be sure to keep the tungsten electrode from coming into direct contact with the metal being welded. Otherwise you'll slag (contaminate) the tip of your torch. The very tip will, however, get a little ball on it because of the AC, but that's normal.

The positive aspect of a tig weld is that you can get almost surgical precision with your welds and, if done carefully enough, these welds may need little or no sanding or buffing to get a clean edge.

Overall, it's is easier to work with a spool or wire-fed torch – as opposed to the two-handed tig method – because with aluminum, you have the current-controlling foot pedal to think about, not just what you're doing with your hands.

When you are using a welding device that pumps 50,000 volts through it, a major issue is cooling. A mig welder uses less amperage than a heliarc, and is air-cooled, while the heliarc is water-cooled.

Another difference between a mig welder and a heliarc is that the mig only heats the area you are working on, whereas the heliarc has to get the whole piece hot before it can get the area to be worked on hot enough for welding.

A mig welder more directly heats the filler metal and the area right below it. I reccomend preheating the metal before welding. If it is small enough, you can pop it into

Left: This is a spool gun for mig welding. It is loaded with 4043 aluminum wire, which is used as a filler metal for welding.
Right: This is another type of spool-fed mig welder. This unit uses copper wire, because aluminum wire is too soft to pass through the long length of the hose without jamming up

the oven for a bit to heat it up, or just use the torch to warm it. Also, when doing a weld, making a "V" of the edges to be welded together is a good idea. This will insure the weld will remain solid through the finishing process in which the upper rough layer is removed to get that nice smooth edge.

This should be done on both sides of the metal, even if the underside or "bad" side isn't going to be finished. It will ensure that enough weld will be there to keep the assembly structurally sound. This is most crucial when doing joint connections.

Bolting

When you have alloys that don't weld well, like the 7000 series and some of the 2000s, you must use bolts to make connections. For example, 7075 cannot be welded; it is almost as hard as mild steel and is not bendable. Series 2024 is bendable and fabricates well. It is commonly used in airplanes.

So, if you need to make a connection to 7075 or 2024 aluminum pieces, bolting is the best way to go. Here is one recommended procedure: First, countersink the holes top and bottom. Second, put threaded inserts in the boltholes. These inserts will often have an activated epoxy on the threads so when you put your screw into the insert, the screw will be securely locked in place. This will prevent fasteners from coming loose due to "creep."

Creep occurs when the aluminum metal moves, or creeps away, from under the bolts as the metal moves, as it would if being used on a car or a robot. Such movement can cause bolts to loosen and fall out or break.

A good example of how the thread inserts work is this: Let's say you have a 1/4-inch, 20-threads-per-inch tap that you want to put into your aluminum sheet. This requires a hollow thread insert that has an outside diameter of 3/8-inch with 16 threads per inch – and an inside diameter of 1/4 inch with 20 threads per inch. You drill out a 3/8-inch hole in the aluminum and screw the insert into your hole in the metal.

This insert is often coated with an epoxy on it, which are microscopic beads of glue that grip onto the outside screw threads and bind them to the metal. Next, tighten a screw into the inside of the expoxied insert to make the connection, and within 24 hours, you will have a secure fit. (The epoxy takes 24 hours to set fully.)

This connection will hold very securely. It is the type of connection used in car engines. If it will hold under that kind of stress, you can be sure it will hold up under the stress of most robotics projects.

One last thing to remember – don't use cheap, low-quality connectors. I found that out the hard way. The low-end connectors (rivets) I used on *Spike II* broke or fell out. The better quality ones – the ones that were a dollar each and designed for aviation use – were the ones that held.

Also, I didn't use inserts and did lose some of the bolts due to creep and the stresses of combat. Even though the robot held together for the entire competition, much of the loss of control and drivability that occurred during the final battles happened because of loose and missing connections. RS&T

Author's Recommendations

Get acquainted with your local metal shop. (Thank you, Frank and Central Metal Fabricators Inc. of Red Bank, NJ, for all the help with this article and with my robot warrior, *Spike II*.)

Central Metal Fabricators, Inc.
34 Willow St,
Red Bank, NJ 07701
tel: 732.530.7339
If you need metal work, talk to Frank Crisafulli.
Central Metal has experience working with all types of metal, including titanium.

Small Parts Inc.
13980 NW 58th Court
P.O. Box 4650
Miami Lakes, FL 33014-0650
tel: 800.220.4242
fax: 800.423.9009
smlparts@smallparts.com
www.smallparts.com/
Great source for miscellaneous parts.
They are a surplus place, so they may or may not have that 6061 aluminum sheet you need for your project, but they have *tons* of stuff you won't find anywhere else.
They also have a great shot of Carlo Bertoccini's *Stealth* robot on the cover of their literature.
Highly recommended!

PEM Fasteners
Danboro, PA 18916
tel: 800.342.5736
fax: 215.766.3633
pem@pemnet.com
www.pemnet.com
If it's a connector, PEM has it.
Their catalog is detailed with engineering specifications.
They have offices worldwide
Check out their Web site!

Silver Bullet, continued from page 35

language by the students. Three sonars face forward, two are mounted on the sides, and one is in the rear of the vehicle to prevent collisions as the robot backs up. Data from the inclinometer (constructed from mercury switches) tells the computer whether a incline is safe to climb slowly, or to back away from it.

Bujold

When *Silver Bullet* can maneuver no closer to a trapped victim, she opens her back-mounted pouch, lowers her ramp and *Bujold* rolls out.

Unlike the custom-built *Silver Bullet*, *Bujold* is a $19,000 off-the-shelf industrial robot from Inuktun Services, Ltd., of Canada (www.inuktun.com/). This tiny but mighty "variable-geometry tracked vehicle" is designed for remote video inspection in confined spaces. Ideal for SAR work.

For simple tasks, *Silver Bullet* can use *Bujold* as a roving eye. For more complex assignments, a human can teleoperate *Bujold* through *Silver Bullet's* control tether. The teleoperator can view *Bujold's* situation from *Silver Bullet's* camera, or get a close-up view from *Bujold's* video feed.

The Researcher

Robin R. Murphy, Ph.D. (magma.Mines.edu/fs_home/rmurphy/), is director of the Mobile Robotics/Machine Perception Laboratory at the Colorado School of Mines (CSM) as well as associate director of CSM's interdisciplinary Center for Robotics and Intelligent Systems. Funds come from the National Science Foundation, the Defense Advanced Research Project Agency, NASA, Colorado Advanced Software Institute and various interested industries.

The laboratory was established in 1992 to create the artificial intelligence technologies needed to remove humans from hazardous situations and to do so in an environment where research, education and technology transfer are inseparable.

"I put a great deal of emphasis on students developing visualization packages and convincing experiments that help communicate the contribution to society of their research rather than on purely theoretical simulation."

Murphy and her students demonstrated *Silver Bullet* and *Bujold* at last year's American Association for Artificial Intelligence Conference. This year, she is co-chair of the 1998 AAAI Mobile Robot Competition (www.aaai.org/Conferences/National/1998/Robots/). Murphy is also faculty adviser to CSM's student AI Robotics Teams.

See: www.mines.edu/fs_home/rmurphy/mrmp/ and www.mines.edu/research/cris/.

RS&T

Pathfinder, continued from page 39

There were, however, several potential problems with enabling priority inheritance on the meteorological data thread semaphore.

First, there was no way to enable the inheritance parameter only on the semaphore for the meteorological thread. Enabling the meteorological semaphore would enable all of the other semaphores in the program as well. How would this change the behavior of the rest of the system?

Second, Wind River had deliberately turned off the priority inheritance option before launch to optimize the performance of VxWorks. Would performance be degraded if it were turned on?

After intense consultations with Wind River personnel and extensive testing on the duplicate *Pathfinder* in the JPL lab, the flight software team determined that there would be no adverse impact by activating the priority inheritance function.

Repair by Remote Control

"Patching " software on a spacecraft on another planet is a somewhat specialized process.

As Reeves explained it, the team had to transmit the differences between what was onboard *Pathfinder* and what was onboard the reconfigured replica in the lab. They used custom software ("with a whole bunch of validation") that was already onboard *Pathfinder* to modify the VxWorks operating system.

The ending was somewhat anticlimactic. Buried in a July 21 JPL press release was this aside: "[The flight software team] also sent a software update to correct sequences onboard the flight computer which have caused it to automatically reset itself."

When asked for any final comments on the priority inversion problem, Reeves said, "Even when you think you've tested everything that you can possibly imagine, you're *wrong*."

Although he broke the news on the program malfunction, Jones stressed, "I greatly admire what the *Pathfinder* team accomplished."

At JPL, Reeves echoed the same sentiment: "The team, dedicated to a single purpose, is what got this thing done. They're the ones who really deserve the credit for making it happen. The Wind River guys did a really good job." RS&T

Anatomy of a Priority Inversion

Lest we've proved ourselves to be "generally incoherent," or worse yet, "wildly wrong," we refer you to Michael Jones' actual report at: research.microsoft.com/~mbj/Mars_Pathfinder/Mars_Pathfinder.html.

You can find Glenn Reeves' response at: research.microsoft.com/~mbj/Mars_Pathfinder/Authoritative_Account.html.

For background on Wind River's VxWorks on *Pathfinder*, go to: www.wrs.com/products/html/jpl.html.

And for information on the entire *Mars Pathfinder* mission, go to: mars.jpl.nasa.gov/default.html

Here's a little of what you'll find in Reeves' analysis:

The software to control the 1553 bus and the attached instruments was implemented as two tasks. The first task controlled the setup of transactions on the 1553 bus (called the bus scheduler or bc_sched task) and the second task handled the collection of the transaction results i.e. the data. The second task is referred to as the bc_dist (for distribution) task. A typical timeline for the bus activity for a single cycle is shown below. It is not to scale. This cycle was constantly repeated.

```
   |< ------------- .125 seconds ------------------------>|

   |<***************|                  |********|   |**>|

                    |<- bc_dist active ->|   bc_sched active
   |< -bus active ->|                            |<->|

----|---------------|--------------------|--------|---|---|-------
    t1              t2                   t3       t4  t5  t1
```

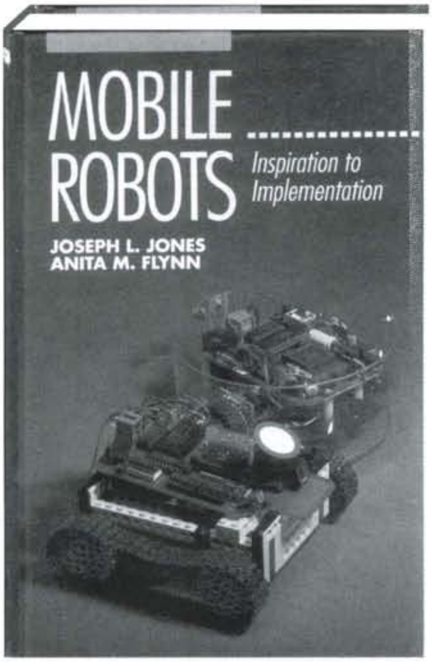
The *** are periods when tasks other than the ones listed are executing. Yes, there is some idle time.

t1 - bus hardware starts via hardware control on the 8 Hz boundary. The transactions for the this cycle had been set up by the previous execution of the bc_sched task.
t2 - 1553 traffic is complete and the bc_dist task is awakened.
t3 - bc_dist task has completed all of the data distribution
t4 - bc_sched task is awakened to setup transactions for the next cycle
t5 - bc_sched activity is complete

Feed Your Head

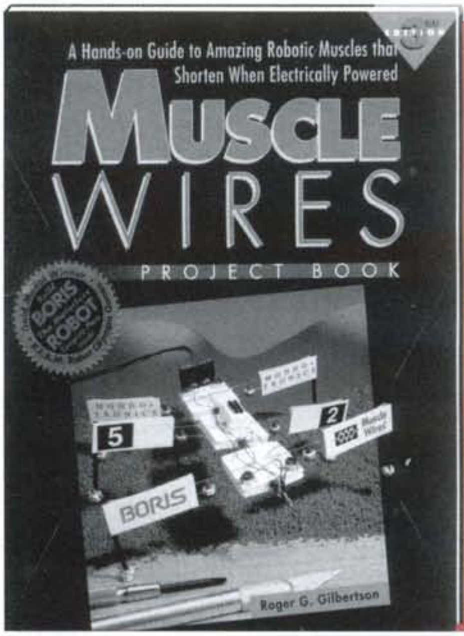
Mobile Robots: Inspiration to Implementation

by Joseph L. Jones and Anita M. Flynn



MIT's Jones and Flynn wrote the bible for small robots, and this is it. Simply stated, "Our aim is to teach you how to build a robot." Yup, they did it. In fact, Jones and Flynn's book opened an esoteric world to us, and made it seem simple. Using the LEGO-based TuteBot and Rug Warrior designs, this book shows you not just *how* to build an indoor mobile robot, but also explains *why* each component works. Hundreds of drawings, diagrams, tables, definitions and photographs illustrate the narrative. Don't be intimidated by the programming code. Just start at the beginning, and by the end, it all becomes clear. The authors give us lessons in microprocessors, batteries, interfaces, light and sound sensors, motors, and mechanics. This book contains some of the most extensive appendices we've seen to date.

Recommended source: A. K. Peters, tel: 617.235.2210, www.akpeters.com
A. K. Peters, ISBN 1-56881-011-3, 349pp hardback, $**48.00**



Muscle Wires Project Book:

A Hands-on Guide to Amazing Robotic Muscles That Shorten When Electrically Powered

by Roger G. Gilbertson

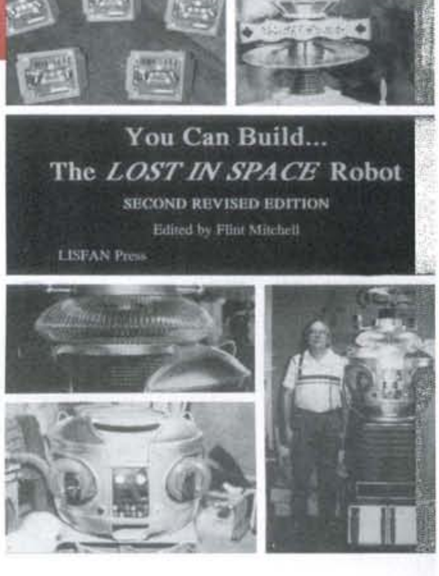
Award-winning entrepreneur and engineer Roger Gilbertson leads us through the mystery of motions without motors. Shape Metal Alloy, in the form of Flexinol wires, provides the basis for 15 detailed construction projects, including a small, motorless, six-legged walking machine. We recommend this book because it is easy to read (without a college education), abundantly and beautifully illustrated, contains a lot of pertinent background theory, and thoroughly discusses the practical applications of Muscle Wires.

Get the book with the "deluxe" package that includes one-meter spools of wire, in varying gauges. Read the theory, study the soldering section, then jump into the simple but elegant projects. No prior knowledge of electronics is needed to enjoy this learning adventure.

Recommended source: Mondo-tronics Robot Store, tel: 800.374.5764 www.robotstore.com
Mondo-tronics, ISBN 1-879896-16-8, 130pp paperback, $59.95

You Can Build...*The Lost in Space* Robot,

edited by Flint Mitchell



LIS captured the imaginations of a generation in 1965, and the robot (which never had a real name) continues to inspire imitation. ***LIS*** fan extraordinaire Flint Mitchell, in collaboration with builders John Rigg, John Burkhalter and others, gives us a firsthand account of how to build your own ***LIS*** robot from scratch. Some simple circuits, some wood and plastic, and presto! Many hours (months) and dollars later, you'll have the satisfaction of building your own life-size replica (with your own added improvements, perhaps). This book suggests sources for parts and supplies, and details working with plastic molds and vacuforming.

Recommended source: LISFAN Press, 7331 Terri Robyn Dr, St. Louis, MO 63129, tel: 314.846.2846, lisfanedtr@aol.com
LISFAN Press, www.as-inc.com/lisfan/lisfan.htm, 98pp paperback, $**9.95**

submission guide

Robot Science & Technology

WANTS YOU!

... to write for us.

Here are some tips on what turns us on (and off):

hot leads, cool conclusions – Make us want to read your article – and be glad we did. Your first sentence should suck us in, and your last sentence should kiss us off.

let's get organized – We want to see an engaging introduction, an interesting narrative, and a satisfying conclusion. Subheads are useful in organizing longer stories.

let's get technical – Tell us exactly how you built and programmed your robot. Where did you get your parts? Cost? Which websites were helpful? Above all, get your facts straight!

let's get personal –We're always looking for the human side of robotics. We'd like to see some robot stories involving family or team efforts.

do it with style – We're looking for lively articles with high readability. Tell us your story in your own words.

be original – Please, no simultaneous submissions or reprints. We want exclusive stories.

get the picture – We want to see crisp, clean photographs and graphics. Of course, we want to see the whole robot, but we also want close-ups of the drive train, power plant, sensor array, or whatever else that is particularly fascinating about your robot ... including you, your family or your team.

stay within the parameters – Articles should be 1,000 words to 2,500 words in length. Email submissions are preferred.

query? – Unless you're a professional writer with published credits, please, do not send us a query letter. We don't want to know if you can write a letter; we want to see that you can write an article.

pay off – We pay on acceptance. How much we pay is directly related to how well you write, and how hot (or cool) your subject is.

Please send your submissions to:
editor@robotmag.com
or:
Editor
Robot Science & Technology
2351 Sunset Blvd., #170-253
Rocklin, CA 95765
USA

For our complete writers' guidelines, visit our website at www.robotmag.com, or send a business-sized self-addressed stamped envelope to the above address.

submission guide

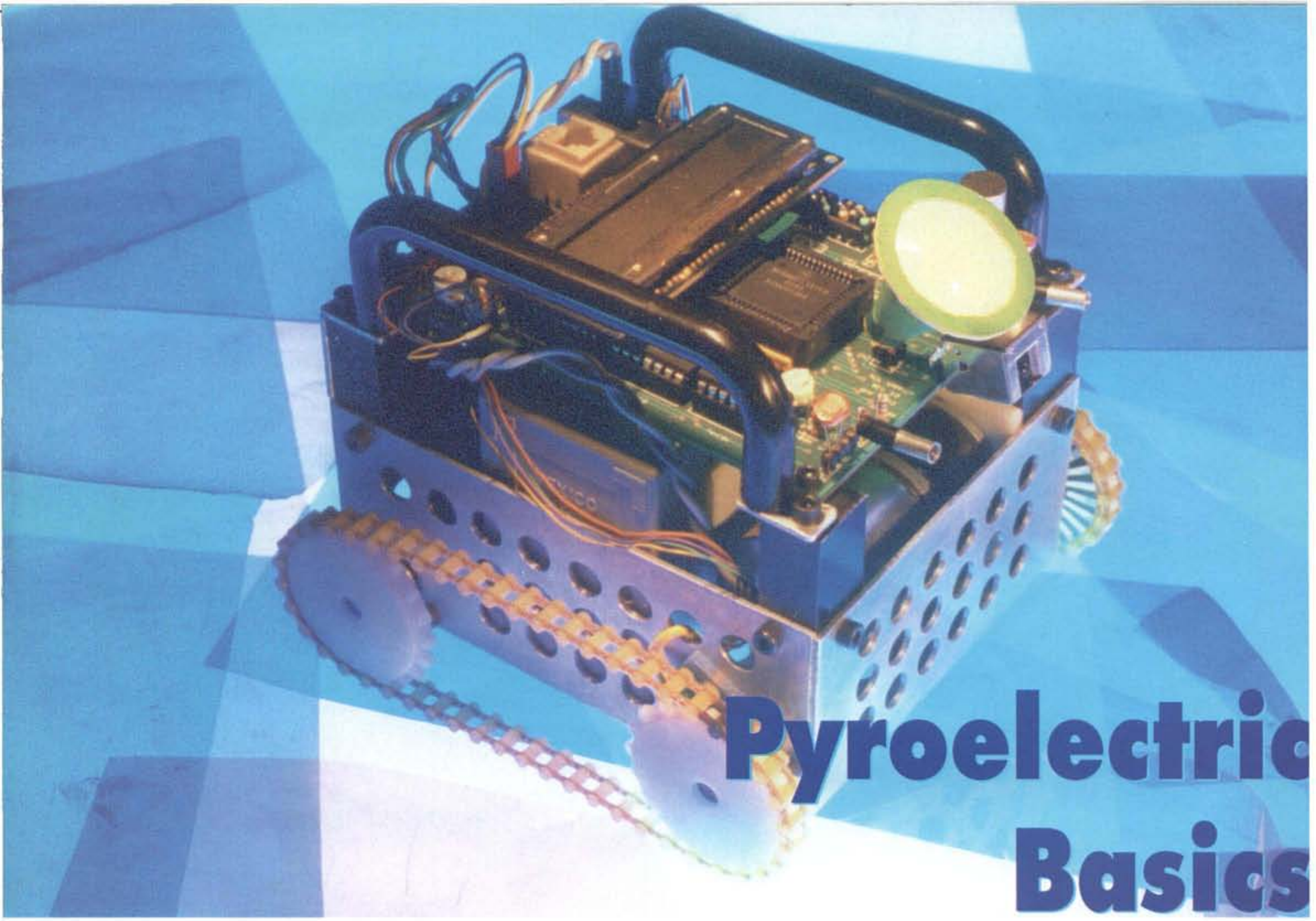
Ladies Tapered Tee $24.95

Ball Cap $19.95

Men's Polo $34.95

Wear our shirt: Get Noticed.

To order call 1.888.510.7728



Pyroelectric Basics

by ***Steve Richards***
Photos by ***Doug Winters***

The pyroelectric sensor is one of the easiest ways to allow a robot to detect the movement of humans and animals. Since this is accomplished through infrared energy detection, pyroelectric sensors can also be used to find other sources of heat such as candles. The most common everyday use of the pyroelectric sensor is in motion-detecting burglar alarms.

Pyroelectric sensors offer passive infrared motion detection and are therefore often referred to as PIR sensors. They are passive in that they don't introduce anything into the environment they sense: they simply observe the environment and measure changes in energy. In contrast, active sensors introduce a signal into the environment. (An example of an active sensor is an ultrasonic ranger that introduces noise into the environment in the form of one or several chirps and then awaits an echo from that noise.)

To understand how a pyroelectric detector works, some background information is helpful.

Humans have a skin temperature of about 93°F (39°C). The emissivity of human skin is around 0.98 times the emissivity of a perfect black body. Also, the infrared radiation from a person is independent of race but can be affected by the inhibited circulation often caused by smoking. This wavelength of maximum energy radiated by humans is about 10 micrometers. (For reference, a light bulb emits peak energy in the wavelength range around 1 micrometer.) The total energy emitted by a typical human is around 800 watts.

Although all humans are naked under their clothes, clothing tends to mask much of the infrared energy emitted. So the job of detecting human movement based on emitted infrared boils down to detecting subtle changes of energy in the 8-14 micrometer range while rejecting changes in other wavelengths that may be caused by light, motors, etc. A pyroelectric detector is designed with this specific task in mind.

The active element in a pyroelectric detector is typically made with a substrate of lithium tantalate crystal, ceramic, or polymer film. In all cases, a slice of the substrate is doped with an electrode on both sides. When infrared energy hits the substrate, heat is generated which displaces electrons, effectively generating a charge between the two electrodes. This small charge is then amplified with an operational amplifier, and the result is a very usable signal that reflects *changes* in infrared energy. Lithium tantalate has the desirable property of being particularly sensitive to infrared energy in the 8-14 micrometer range—perfect for detecting humans. Lithium tantalate is also the most stable substrate with respect to temperature.

A single pyroelectric detector can find changes in infrared energy in the 8-14 micrometer range. One way to improve the signal is to use two detectors placed side-by-side. The voltage difference between the two detectors can be amplified and measured to get a more sensitive reading. Keep in mind that the two detectors must have a slightly different environmental view to notice the change. (Differing viewpoints are often accomplished either by detector separation or optics.) Two detectors housed in one package are known as a dual-detector device. This dual-detector device package often improves noise rejection from large air mass movement, such as wind, and is therefore preferred outdoors.

Typically, the detector is packaged like a transistor in a TO-5 metal can or something similar. The top of the can has a window through which the detector receives infrared energy. That window is often constructed from materials that filter wavelengths outside the desired range. In this way, specific wavelength filtering helps avoid false measurement from lights, IR-rangers or other sources of infrared energy.

This package usually contains three or four pins. The 422-3 is one of the most widely used pyroelectric detectors in robotics. The number 3 in the part number refers to the window material used and is specifically designed to transmit wavelengths from 7-16 micrometers.

The output signal from a pyroelectric detector typically floats near a steady state value. The 422-3 hovers around 2.5 volts for its steady state. When a source of infrared energy in the 8-14 micrometers wavelength moves into the detector's field of view (FOV), the voltage at the output will drop or rise depending on the direction of the infrared energy motion.

If a person walks into the FOV of the detector, the detector will show a short-term, direction-dependent rise or fall. If the person stays in the field of view but stops moving, the detector will stabilize to a relatively constant steady state voltage until the person moves again. The detector, therefore, is not a true detector of human *presence*. It is rather a detector of human *movement*.

Some examples of robotic behaviors that incorporate a pyroelectric detector into their sensor suites might be:

Opposite: Steve's hacked Rug Warrior

Watchdog

Here the robot can stay still in a corner and watch its surroundings. If a person moves nearby, the robot will detect the movement and could "bark" using a speaker or piezoelectric buzzer.

Follow the Human

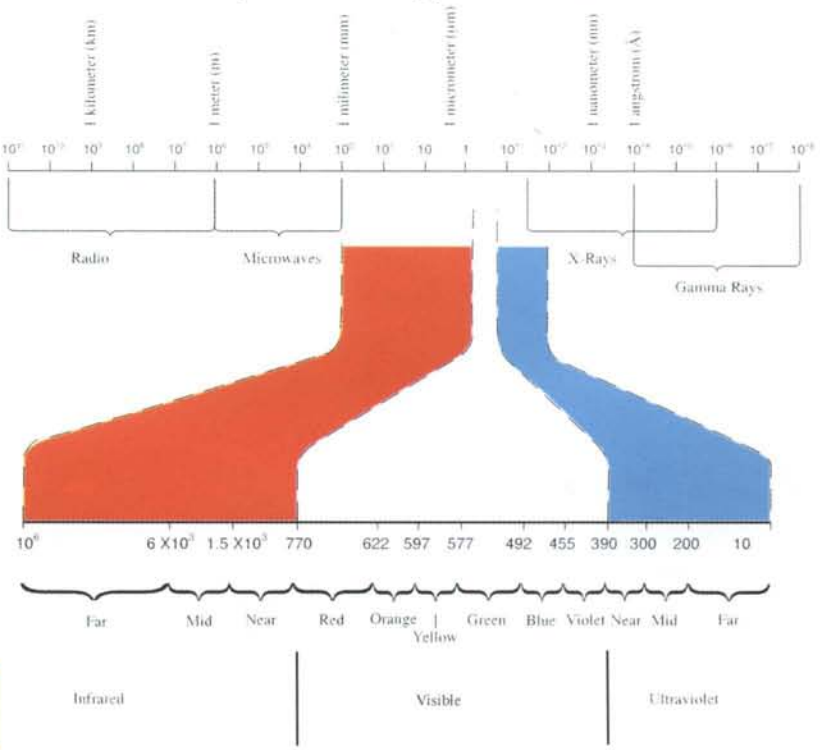
The robot's ability to detect movement allows it to follow the direction of the human. This is made possible through the use of its detector, which captures the movement in its field of view. In this way, the robot will be able to stay in contact with the human. This could also be used to "chase the cat."

Find the Candle

A candle in a room has enough infrared emission in the 8-14 micrometer wavelengths from the carbon in the flame to be detected by a pyroelectric sensor. The candle is not likely to move, however, so the detector must be "panned" around the room until a spike (positive or negative) is detected by the pyroelectric detector. This spike corresponds to the heat generated by the candle. One way to determine if the spike is caused by a candle and not some other energy source is to point the detector directly at the source of the spike. Candle flames flicker at about around 4 to 20 cycles per second. Watching for a change in energy in this frequency range is very reliable when trying to distinguish a candle flame from other sources of infrared energy.

Avoidance

Big robots can hurt people! A software driver could constantly monitor the pyroelectric detector. When the



software driver detects movement around the robot, it can trip an interrupt to stop or slow the robot. The robot can then use a warning buzzer, speech chip or another mechanism to warn the offending human. Once the pyroelectric detector settles down to its steady state, the robot might assume "no moving-human risk" and resume its activities.

The Interface

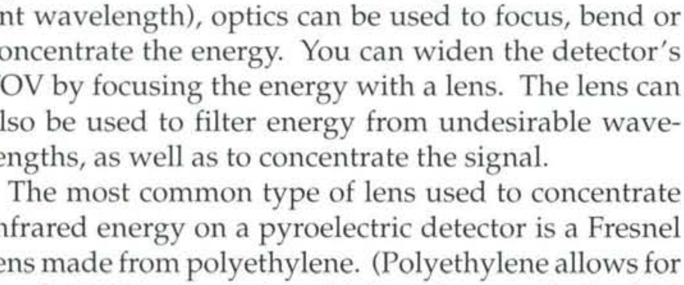
The method of interfacing microcontrollers or logic circuitry to a pyroelectric detector depends on the detector being used. Some detectors, such as the Eltec 422-3 dual-detector, incorporate the op-amp circuitry internally in the detector package. The output signal of such a detector can be read directly with an analog/digital (A/D) channel on a microcontroller. Other detectors, such as those made by Nippon Ceramic, require amplification to be easily read by a microcontroller's A/D port. The detectors with integrated electronics are more costly but offer greatly reduced complexity and power consumption. Below is a block diagram of the internal circuitry in the Eltec 422-3.

If an A/D converter is not available in the controller, two comparators can be used to establish logic output from the detector. The two comparators are adjusted to voltage threshold values above and below the steady state of the detector. In this way, one of the comparators will change state when the energy source moves right-to-left, and the other will change state when the energy source moves left-to-right.

Since the energy (8-14 micrometers) detected by a pyroelectric sensor is like visible light (but at a different wavelength), optics can be used to focus, bend or concentrate the energy. You can widen the detector's FOV by focusing the energy with a lens. The lens can also be used to filter energy from undesirable wavelengths, as well as to concentrate the signal.

The most common type of lens used to concentrate infrared energy on a pyroelectric detector is a Fresnel lens made from polyethylene. (Polyethylene allows for nearly 80% transmission of infrared energy in the 8-14 micrometer wavelength range.) Because a Fresnel lens is segmented, its design provides for a very thin, flexible optic. Fresnel lenses have a focal length just like continuous lenses. The focal length of the lens determines the optimal location for the pyroelectric detector behin the lens.

Block Diagram



Reprint courtesy of Eltec Electronics Inc.

An optical head is typically used to properly orie the lens and detector in a package that can then b panel-mounted to the head or body of the robot. A optical head protects the detector and lens and shiel the detector from ambient interference outside the FO An optical head can be obtained commercially or mac from a cone constructed of stiff card stock. The di tance from the back of the lens to the detector windo should be as close as possible to the focal length of th lens. Commercial sources for optical heads are usuall found through the makers of the lens or the detecto

It's easy to extend the capabilities of a robot with pyroelectric sensor. A detector with internal signal pr cessing circuitry requires only 5 volts, ground and spare A/D channel in your hardware. The fact that th robot can then exhibit rudimentary interaction wi people and animals can add a whole new dimensio to the behavior of your robot.

Pyroelectric Detector Application Hint

The infrared wavelength of 8-14 micrometers does n pass through glass, certain plastics or acrylic. Be caref not to obscure the detector on your rob behind such materials. Be aware that infrared energy use by many IR-rangers will pass through such materials.

Since the energy in humans being detected by pyroelectric detector is infrared energy emitted by th skin, a detector pointing toward the face-level of humar gives the most reliable signal when seeking humans.

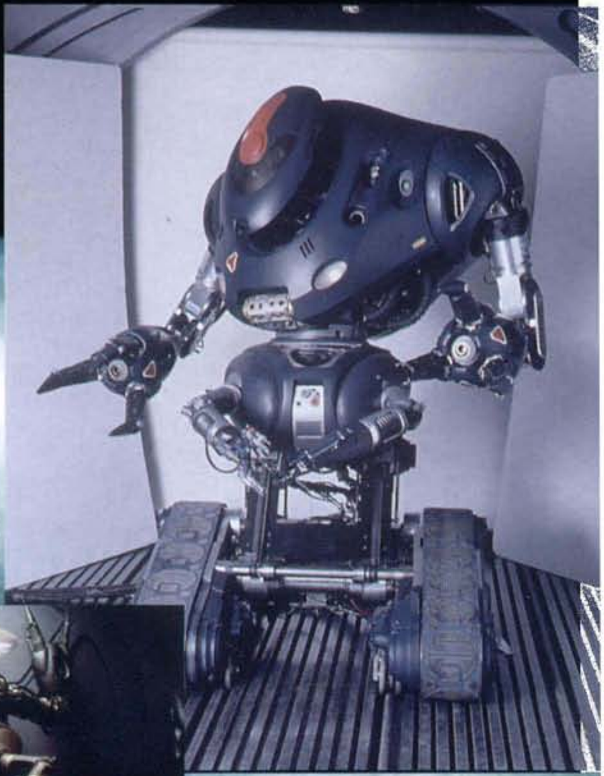
Pyroelectric detectors measure the voltage created b the slight heating of the substrate such as lithium tant late crystal. As a result, it is not a good idea to solder c otherwise heat pyroelectric detectors as would occur i wave soldering. The energy created by the excessiv heating of the substrate can create enough volts in th internal electronics of the detector to blow the detecto

Detectors can have an inverted signal if the source c movement is cooler than that of the background. Co sider the scenario of a detector pointing towards a 105° (41°C) wall. If a human was to move across th

continued on page 6

Lost in Space? Of course, we've got that covered

See Our Interview With Robot Builder Verner Gresty On Page 24



To Our Children's Children

Photos courtesy New Line Cinema

Pyro, from page 60

detector's field of view, the signal would be inverted. This is due to the fact that the temperature of the human is cooler than that of the wall behind him or her. That is to say, a detector that typically rises in voltage when you move right to left in front of it would instead fall in voltage.

Dual-detectors have two elements that must be oriented such that the motion being measured is across the two detectors rather than up and down. The readings will be anything from insensitive to meaningless if the detector is not situated properly. The documentation for the detector in question will likely describe the internal organization of dual-detector devices.

Detectors with internal electronics often have a fourth output pin that is a constant voltage reference. This reference can be used for further amplification of the detector output. Another handy use of this reference voltage is to couple it with a comparator, which will create a low-battery warning sensor. Check the specifications of your detector to see if this is feasible.

Author's Recommendations:

Mobile Robots: Inspiration to Implementation
by Joseph L. Jones and Anita M. Flynn
ISBN 1-56881-011-3

Sensors for Mobile Robots
by H.R. Everett
ISBN 1-56881-048-2

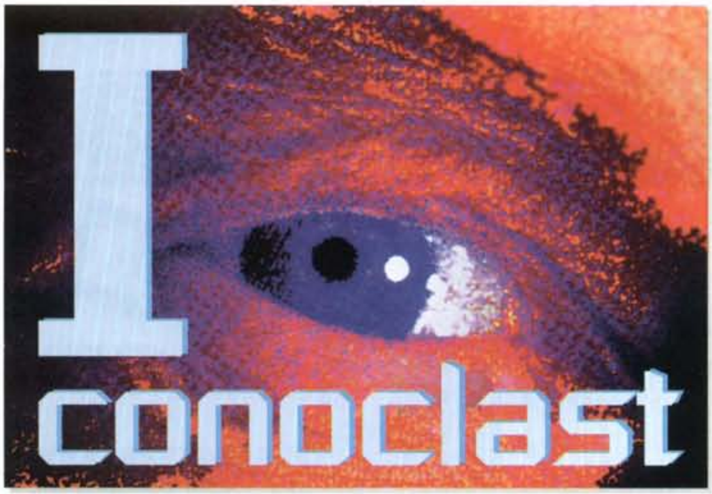
Using Lithium Tantalate Pyroelectric Detectors in Robotics Applications
by David Cima
Eltec Instruments, Inc.

Pyroelectric Detector:

Acroname Inc.
P.O. Box 1894
Nederland, CO 80466-1
www.acroname.com
303.258.3161

Eltec Instruments, Inc.
P.O. Box 9610
Daytona Beach, FL 321
904.252.0411

Scitec Instruments Ltd.
Bartles Industrial Estate
North Street
Redruth, Cornwall
TR15 1HR England
+44.1209.314608
www.scitec-inst.demon.co.u



Home Robots, NOT!

Even if a practical home robot could be built, the maintenance would be a killer — possibly worse than that of a photocopier.

Ask amateur roboticists what kind of robot they would really like to build and most will describe some kind of home robot. Perhaps to vacuum the floor, perform security, fetch a beer, or just annoy the cat. After giving this a lot of thought, I've concluded that building a home robot is not a good idea, especially for hobbyists. By hobbyist, I mean someone whose profession is not robotics and who, therefore, has limited time, resources and technical background to solve the many difficult problems that mobile robots demand.

The home is a cluttered and dynamic environment. There is an unpredictable variety of objects of irregular shape, size and placement. Many objects are fragile and may be added, removed or relocated daily. There are varying floor surfaces and level transitions in the manner of steps and thresholds. So if the home environment is so hopeless, why do so many hobbyists develop their robots to operate in the house?

> **Doors open and close. Pets, small children and unenlightened visitors dash about. What's a robot to do?**

Many people who design a home robot do so for one of two rather different reasons. Some want to sell to an imagined large market. I say imagined market because, although a home robot sounds like an exciting novelty, I doubt many will want to pay a hefty price for one. And think of the product liability potential! An attractive, nicely designed robotic lawn mower has been on the market for a few years and has not yet taken the lawn care industry by storm. Why? Because it's ten times more expensive than a traditional lawn mower, yet is less capable. The same will be true for early home robots. Robots will first have to prove themselves in a more structured environment, such as an industrial setting, before they will be accepted in the home.

The second reason that hobbyists design their robots to operate around the home is simply because that's where the robot workshop is. Notice that university researchers build robots that operate around the lab or university buildings. Anyone who builds robots knows that the process involves endless testing and trips back and forth to the workbench. It's simply convenient to have the workbench next to the operating environment. Only occasionally does anyone design a robot to operate outdoors. It takes a rather large grant, access to a professional machine shop and a full-time staff (or graduate students) to build a Mars *Pathfinder*, a volcano-exploring *Dante*, or an automatically piloted car.

Not withstanding the inconvenience of having shop and operating areas in different locations, I encourage robot builders design for an out-of-home environs. One venue that is better suited for a mobile robot is found in most office buildings. They are usually well-lighted, highly structured, uncluttered and vacant at night and weekends. Hallways tend to be straight and intersect at right angles, doors are of a uniform size, the floor is level and evenly surfaced, etc. Yet there are more than enough variations to still make it a challenging job. Other possibilities are warehouses or factory floors. Many of these also have a regular, uncluttered environment and so have market potential as well. If you are *really* interested in an *outdoor* robot, then I suggest one designed to operate on the surface of a lake!

Robust mobility is a difficult challenge in any environment. The home requires that all of a robot's resources be concentrated on overcoming obstacles and moving objects instead of performing interesting or useful tasks. To me, the point of a mobile robot is more than mobility itself. By choosing less cluttered environments, robot designers can build their skills before attacking their homes.

APRIL 1, Tech Challenge '98

San Jose, Calif. • www.thetech.org

More than 200 student teams from Silicon Valley and the San Francisco Bay Area converged in the San Jose Civ Auditorium April 1 for the 11th annual Tech Challenge. RS&T was there.

April 2-4, 1998 FIRST Competition

Orlando, Fla. • www.usfirst.org

For Inspiration and Recognition of Science and Technology, the national FIRST Competition takes place at the EPCOT Center at Walt Disney World with more than 165 teams of high school students, along with their university and corporate sponsors.

April 18, UC Davis Picnic Day MicroMouse Competition

Davis, Calif. • www.ece.ucdavis.edu/umouse/

This wide-open competition has fully autonomous "micromice" racing through a 12x12-foot maze. This unofficial scrimmage is fun practice for the upcoming official IEEE Region 6 micromouse competition May 2.

April 18-19, Trinity College Fire Fighting Home Robot Contest

Hartford, Conn. • www.trincoll.edu/~robot/

Strictly autonomous, home-built robots must negotiate an 8-foot-square model house, find a burning candle, and extinguish the flame. See pages 4, 5, 9 and 58 in this issue.

April 30 – May 2, SAE National Robotics Walking Machine Decathlon

Northern Illinois University, DeKalb, Illinois • www.sae.org/ns-search/STUDENTS/

One of the toughest walking robot competitions in the nation, the Society of Automotive Engineers. **Walking Machine Decathlon** requires college teams to put their robots through 10 tests, including a dash, avoiding a crevasse, object seeking, object retrieval and a hill climb.

May 1-2, SME Student Robotics Automation Contest

Saginaw Valley State University, Saginaw, Mich. • www.sme.org

The Society of Manufacturing Engineers **Student Robotics Autonomous Contest** offers nine different contest categories. Categories range from stationary Automation Work Cell and Pick-and-Place robots to mobile Sumo and remote-control vehicle robots.

May 9-14, WAC-FIRA American Cup '98

Anchorage, Alaska • www.fira.net/fira/

The Federation of International Robot-Soccer Associations will stage competitions for MiroSot and NaroSot robots in cooperation with the World Automation Congress '98. Winners will compete at the **FIRA Robot World Cup France '98** in Paris, June 29 – July 3.

May 30-June 1, AUVSI Autonomous Ground Robotics Competition

Rochester Hills, Mich. • avdil.gtri.gatech.edu/AUVS/

Small, unmanned and completely autonomous motor vehicles must negotiate an outdoor obstacle course – rain or shine. Sponsored by the Association for Unmanned Vehicle Systems International.

events

MONDO•TRONICS' ROBOT STORE™

YOUR SOURCE FOR ROBOT KITS, BOOKS, PARTS, TOOLS, VIDEOS, ELECTRONICS, MUSCLE WIRES & MORE!

ROBOT KITS KITS KITS!!!

Hyper Peppy Robot #3-296 $29.95
Bright red, white & blue with clear dome. Runs forward until it bumps object or hears loud noise, reverses & turns, then resumes. With complete instructions. 14 cm W x 13 L x 9 H. No soldering required. Requires 2 AA cells (sold separately). *A fun & frantic first robot!*

Sonic Rover Robot #3-174 $29.95
A great, low-priced kit. "Bump & run" operation like Hyper Peppy above. Molded body, red/green LED, great 48 pg manual. 14 cm. L x 12 W x 7 H. Requires soldering and 2 AA cells (sold separately). *A fast moving first robot!*

Spider Walking IR Avoider Robot #3-299 $69.95
Futuristic & maneuverable 6-legged walker with IR for object avoidance. Dual motor drive, front mounted sensor. 13 cm L x 13 W x 10 H. No soldering required, needs 2 AA & one 9V cell.

Soccer'bot Kit #3-391 $34.95
Fast, Furius fun. These spunky little six-legged walkers move up to 6 meters / minute. Wired remote control of forward, back and rotate! 12 cm L x 9 W x 8 H. Needs no soldering! Each kit (one 'bot per kit) requires 2 AA cells.

Cybug Robot #3-466 $49.95 NEW
Cybug has touch and light sensors to exhibit active and interesting behaviors. Printed circuit board & 2 DC motors! 12 cm W x 10 H x 4 L. Soldering and one 9V cell required. Machine or living organism?

ADVANCED ROBOTS

Explorer 'Bot Deluxe Kit #3-431 $229.00 NEW
Develop your custom robot hardware and software easily! Includes Basic Stamp 1, prototyping board, two servos, drive-wheels, front caster and a great manual on diskette with full instructions and cool projects! 16 cm L x 11 W x 9 H. Requires soldering & 4 AA batteries. Start creating advanced robot intellegence with this kit!

WAO II Programmable Robot #3-280 $74.95
A sophisticated, full function robot kit. 2 motors, programmable 4-bit micro, 26 button keypad, sensor arms for collision or edge detection, Moves in all directions, performs if-then decisions based on sensor input, even do-loops! Requires soldering, 3 AA & one 9V battery. Great intro to robot programming.

PC Interface for WAO II #3-195 $29.95
Parallel cable & BASIC DOS disk for easy program handling.

New Version of a Top Selling Kit!
Rug Warrior Pro Deluxe Kit #3-406 $595.00 NEW
A complete, serious research robot platform from MIT. With all hardware, C software, 68HC11 CPU, memory, sensors, wheels, motors, chassis, instructions & more. 18 cm dia. The ideal companion kit to the Mobile Robots book (3-098 above). *Mac & PC compatible!*

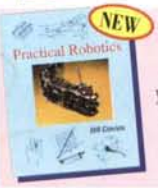
Robot Builder's Bonanza #3-114 $18.95
Practical, ready-to-use circuits, devices, mechanisms, sources, tools, vision, grippers, navigation & more! 336 pgs, illust.

Gadgeteer's Goldmine #3-129 $24.95
Unique high-tech projects & hard-to-find info. Robots, high voltage Tesla & Van de Graaff generators, Kirlian photos, lasers & much more! 400+ pgs, illust.

Mobile Robots #3-098 $47.95
From MIT, a complete guide to designing & building robots: hardware, software, sensors, motors, "subsumption" software. 336 pages, illust. A must for all roboticists.

Sensors for Mobile Robots #3-145 $67.95
Whoa! Everything about sensors: tactile, proximity, range, acoustic, laser, radar, microwave, path following, dead reckoning, compasses, gyros & more. 550 pgs. Fun & intelligent!

Practical Robotics #3-416 $39.95 NEW
If you ask "where do I start?" the answer is "HERE!" Bill Davies shares years of robotics experience, covering all aspects: tools, techniques, components, sensors, soldering, etc... 330 pages with a fine index.



IC Hobbyist's Handbook #3-164 $19.95
Toss those old IC data books! This practical collection has 80 common ICs to make 100s of circuits! Amps, linear, logic, CMOS – with specs, pinouts, sample circuits – even component values (gasp!). 120 pgs. A basic resource!

Controlling the World With Your PC #3-165 $35.00
Convert ordinary serial & parallel ports into object-detecting, motor-spinning monsters! 42 circuits AND software in BASIC, C and Pascal! 256 pages, illustrated, with DOS disk.

Build Your Own Underwater Robot #3-418 $29.95 NEW
Dive in to underwater robotics! The authors provide dozens of hands on experiments using readily available parts. 148 pages. Great illustrations and a lot of fun. Dive in!

Want to see more? Visit our expanded website to get information and pictures for 100's of robot related products and much more!
Also... Request a free printed catalog online!
www.RobotStore.com

ORDER TOLL FREE — 24-hr Order Line - Visa / Mastercard / Discover
800-374-5764

MONDO-TRONICS
4286 Redwood Hwy. #226-135
San Rafael, CA 94903
Questions: 415-491-4600
Fax: 415-491-4696
Email: info@mondo.com
Web: www.RobotStore.com

Send check, MO or credit card info. All orders add shipping:

Subtotal	Shipping
$25	$6.00
$25 to $50	$7.00
$50 to $75	$9.00
$75 to $100	$10.00
$100 to $250	$12.00
Over $250	$15.00

Calif. orders also add 7.25% sales tax

International Orders Welcome! See website for details

BASIC STAMPS

Stamp One Starter Pack #3-444 $99.00
A complete starter set: Stamp Software, manual, cable, a Stamp 1 (see below), and a BS-1 Carrier Board with prototyping area & connector for the programming cable. A great value!

Stamp Two Starter Pack #3-439 $148.00
Everything needed to move with the Stamp 2: All of the items in the Stamp One Starter but with a Stamp 2 and BS-2 Carrier instead. Monstrous power!

BASIC Stamp 1 #3-201 $34.95
8 input/output lines, EEPROM for 100 instructions, 4 MHz.

BASIC Stamp 2 #3-198 $49.00
16 input/output lines, EEPROM for 600 instructions, 20 MHz clock (up to 38,400 baud), touch tones, X-10, more!

Mini SSC 2 #3-448 $48.95 NEW
Control up to 8 servos via one serial line. Talk to it from any PC, Mac, Stamp, etc... Plus, daisy chain up to 32 together to control up to 255 servos. A must for animatronics!

LYNXMOTION PRODUCTS NEW

5 Axis Robotic Arm Kit #3-366 $195.00
Fast, repeatable, and controlled from your computer! Can lift over 160 grams. Control swivel, shoulder, elbow, wrist, and gripper. Mini SSC and DOS software included - software upgradable!

Hexapod Walker Kit #3-434 $150.00
Three servos provide full directional walking. Programmable from your PC. All hardware, electronics, software and instructions included. Over 20 cm long. Soldering required.

Mega Hexapod Walker Kit #3-423 $375.00
It's huge! 12 servos drive this platform with full control and major strength. Add your own Mini SSC2's, a controller (Stamp2, BOTboard) and some sensors for advanced autonomous behavior!

IRPD Vision Kit #3-464 $29.95
Give your robot the gift of Vision! This Infra-Red Proximity Detector can detect an object from 10 - 30 cm away! Connect to a Stamp, HC11, PC parallel port, or your own circuits.

MUSCLE WIRES & SMA'S

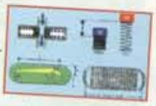
Muscle Wires Project Book & Deluxe Kit #3-168 $59.95
Create direct linear action with Muscle Wires. They contract up to 5% when powered! Use them in robots, models and anything needing strong all-electric *motion*. 128 page book, full plans for BORIS, a motorless walking robot & 14 others. Includes one meter each 50, 100 & 150 µm dia Muscle Wires. 128 pgs, illust. *Get moving today!*

Muscle Wires Project Book only #3-133 $17.95
Get the book now, order wires later!

Space Wings Kit #3-001 $19.95
Sleek silver wings flap using only 5 cm of Muscle Wire! Variable speed up to 36 flaps/minute. 15 cm tall. With circuit board, all parts & instructions. Requires soldering & 2 AA cells. Annoys cats! *An ideal introduction to Muscle Wires.*

Shape Memory Actuator Design Manual #3-092 $69.00
Definitive engineering knowledge about SMAs packed with theories, equations, & references for designing springs, torsion & cantilever devices & more! 160 pages, illustrated.

Shape Memory Design Software #3-139 $299.00
Now! Interactively solve a wide range of SMA design problems: helical springs, torsion & cantilever devices, linear actuators, more. Work in metric or U.S. Fine-tune your design, then print, graph or save results to disk. 45 pg manual. For Windows. Powerful!



Check out "Muscle Wires to Mars" on our Website